Deciding among Gain Scores, Ancova, and Propensity Matching Procedures

Abstract

Researchers have several options available to analyze data from interventions when participants have not been randomly allocated into conditions. Among these are the gain score, Ancova, and propensity matching procedures. Each of these attempts to account for differences among the conditions, but they do so differently. After reviewing these procedures, causal models are hypothesized for how data for an intervention may arise, the data simulated many times, and these procedures applied to each of these replications. For some situations the gain score procedure produced biased estimates (e.g., when the covariate influences group allocation) and the Ancova and propensity matching procedures produced less biased results. Other situations (e.g., when the covariate did not influence treatment) the opposite occurred. For the conditions tested, the propensity matching procedure outperformed Ancova. The main conclusion is that models should be hypothesized for how the data may arise, data simulated for these models, and the properties of statistical procedures evaluated. None of these procedures will provide accurate estimates for all situations.

Keywords: Propensity Matching, Ancova, Gain scores, Lord's paradox, Causality, Graphs, Simulation

Deciding among Gain Scores, Ancova, and Propensity Matching Procedures

When researchers propose a new treatment, it is important to evaluate whether it is beneficial. If people have been randomly allocated into conditions researchers can compare the treatment group with a control group on some outcome variable. In much psychology it is not possible to allocate people randomly into conditions: a health psychologist cannot force someone to be pregnant; a neuropsychologist cannot injure someone; an economic psychologist cannot force a company to commit fraud, *etc.* This is true in many disciplines (e.g., an astronomer cannot assign a star to go supernova; a historian cannot randomly decide whether Trotsky or Stalin succeeds Lenin; see also the evaluations at `www.povertyactionlab.org`).

Because of this researchers rely on statistical techniques with the belief that these techniques somehow "control" the effects of other variables, usually called covariates. These procedures do not physically "control" or in any other way effect these covariates. Unfortunately this falsehood has misled generations of students. Three of these procedures will be discussed: gain scores, Ancova, and propensity matching. All of these provide accurate solutions in certain circumstances to certain questions. The difficulty is knowing when each of these, and if any of these, is appropriate. These methods are described in more detail. Graphical models and simulation are used to explore when they produce biased estimates.

Example: Evaluation when there is a collider

An example intervention will be used throughout most of this paper. It was chosen to represent evaluations where the researcher wants to know if a treatment works, has a prior score measured before the treatment that is on the same scale as the outcome measure, but cannot randomly allocate people into a treatment and a control condition. There are many issues researchers face. The focus for this example is on colliders, which are explained below. It is important to stress, however, that the basic steps:

1. describe models for how the data might arise,
2. simulate data according to these models, and
3. evaluate different statistical procedures,

can be applied to all interventions.

Suppose you want to assess whether year-long weekly after school math clubs improve students' scores at the end of the year. Denote being in the club with $Treatment = 1$ (0 otherwise) and the end of year assessment with *PostTest* (subscripts will not be used on variables in text or in figures, but all vary by participant). There would be complaints if students were randomly assigned to this intervention, so instead students volunteer for it. Who volunteers for an after school math club is not random. Assume there is some latent variable, call it *Propensity*, that predicts whether someone volunteers. *Propensity* is likely also to influence many other things, including how much math knowledge the student has already learned. Call this *Knowledge*. *Knowledge* will also be affected by other constructs including the student's intelligence, effort/grit, home environment, *etc.* Call this collection of constructs *Other*.

Graphs, Causal Models, and Colliders

It is often useful to draw relationships as a graph (not everyone agrees, see Imbens & Rubin, 2015, p. 22). A graph, in its mathematical sense, is a set of nodes, some of which are connected by edges. In causal models these edges are often shown with an arrow on one end to denote the direction of causality. The classic reference applying graphs to causal models in science is Pearl (2009), and good introductions include: Elwart (2013), Morgan & Winship (2007), and Pearl et al. (2016). Figure 1 shows the model described above (each variable also includes an error term, which are not shown in order to make the figures less cluttered). Models are simplifications. For example, it may be that *Propensity* and *Other* are causally related, but this is not shown here. The primary interest is estimating the edge $Treatment \rightarrow PostTest$, enclosed with a red ellipse. This is the direct path. There are

also backdoor paths between these nodes. These are:

1. *Treatment* ← *Propensity* → *Knowledge* → *PostTest*
2. *Treatment* ← *Propensity* → *Knowledge* ← *Other* → *PostTest*

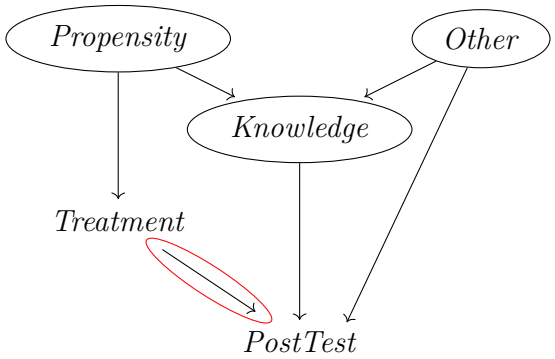

Figure 1. The data model of the latent variables, outcome variable, and treatment. The direct effect from *Treatment* to *PostTest* is enclosed in the red ellipse.

Paths can be either blocked or unblocked. If they are unblocked it means information can flow along them confounding measurement of the direct path. Therefore, often a goal of choosing covariates is to block backdoor paths. How detrimental the effects of an unblocked door path are depends on the product of the path coefficients (Loehlin, 1998). To understand blocking paths it is necessary to consider three ways in which three variables, call them X, Y, and Z, can be causally related:

Chain: $X \rightarrow Y \rightarrow Z$,

Fork: $X \leftarrow Y \rightarrow Z$, and

Collider: $X \rightarrow Y \leftarrow Z$.

Pearl (2009, pp. 16–17) describes two rules to determine if the path between two nodes, X and Z, is blocked and how this is affected by conditioning on the middle variable Y.

Pearl's first rule is that if a path contains a chain or a fork, it is unblocked unless the middle variable is conditioned upon. Chains and forks are associated with the phrases mediation and spurious correlation in the psychology literature. An example of an

unblocked chain is that an exercise pamphlet can lead to students planning to exercise and this can lead to more exercise (Hill et al., 2007). If you prevent students from the planning phase then giving participants a pamphlet is not as effective. A common textbook example of a fork is the positive association between ice cream consumption and murder in cities (Peters, 2013). The warm weather causes both of these to increase. If you could condition on the weather by looking just at one particular weather (e.g., sunny and 83°F), this would block the path and assuming no other effects are present, the correlation would be near zero. In Figure 1, the path $Treatment \leftarrow Propensity \rightarrow Knowledge \rightarrow PostTest$ includes a chain (middle variable *Knowledge*) and a fork (middle variable *Propensity*) and therefore begins unblocked. Much of the justification for using covariates is to block paths like this. One way to block this path would be to condition upon *Propensity* or *Knowledge*, though because these are latent variables it is not possible to condition upon them. Instead researchers use measured variables that are associated with them. Unless the association between the latent and observed variable is perfect, this conditioning does not eliminate the flow of information (so does not completely block the path), but does lessen the amount of information flow.

Pearl's second rule is that a path with a collider begins block, but is unblocked if the collider, or any variable influenced by it (called *descendants* in graph theory terminology), is conditioned upon. Colliders are less discussed in the psychological literature than forks and chains. Wright (2017) uses a river metaphor to describe a collider. Imagine two tributaries arriving from different directions at a deep sink hole. The hole is the collider. The water from each would not reach the other; the path is blocked. If the hole is filled, water could flow between the tributaries and the path would be unblocked. A common example of a collider affecting the measurement of a direct effect is the smoking-birth weight paradox (e.g., Pearl et al., 2016; Wright, 2018). This paradox refers to the finding that smoking appears to *lower* infant mortality rates *if* you condition on the infants' weight. The reason for this is that low birth weight is influenced by smoking, but it is also

influenced by a lot of other things, some of which are much worse than smoking with respect to infant mortality. Therefore, conditioning compares infants with low birth weight due to smoking with infants with low birth weight from these other causes, and smoking is relatively less detrimental. Birth weight is a collider in the path $Smokes \rightarrow LowBirthWeight \leftarrow Other \rightarrow Mortality$ because it is influenced by both smoking and other causes. In Figure 1 the backdoor path $Treatment \leftarrow Propensity \rightarrow Knowledge \leftarrow Other \rightarrow PostTest$ has a collider, *Knowledge*, so this path begins blocked. Unfortunately one method for blocking the first path (conditioning on *Knowledge*) will unblock this path and conditioning on descendants of this collider will also unblock this path.

In the typical social and behavioral science project, particularly one where random allocation is not possible, it is common to measure several other variables with the hope of blocking (and keeping blocked) all backdoor paths thereby isolating the direct effect. Here it is assumed that there are three measured variables that are related to *Propensity*, three to *Knowledge*, and three to *Other*. For the simulation each of these observed variables is correlated $r = .50$ with their associated latent variable, but how they are associated with the latent variable differs. One influences the latent variable (e.g., $p1 \rightarrow Propensity$), one is influenced by the latent variable (e.g., $Prior \leftarrow Knowledge$), and for the other in the trio another variable (depicted just with a circle) influences both (e.g., $o3 \leftarrow \bigcirc \rightarrow Other$). The corresponding graph is shown in Figure 2. While this model looks complex, like all models in social science it is still a simplification. Several of the nodes listed likely influence others (e.g., $Other \rightarrow Propensity$), there are many other constructs that play a role, and there are many other variables that could be measured. After the main simulation some extensions will be considered. It is worth noting that observed variables are not placed along the paths from *Treatment*, *Knowledge*, and *Other* to *PostTest*, and there are no nodes just affecting *Treatment*. These effects could provide additional ways to measure the treatment effect. There are several ways to address measurement of interventions and readers are

encouraged to consult textbooks devoted to this (e.g., Imbens & Rubin, 2015; Morgan & Winship, 2007; Pearl et al., 2016).

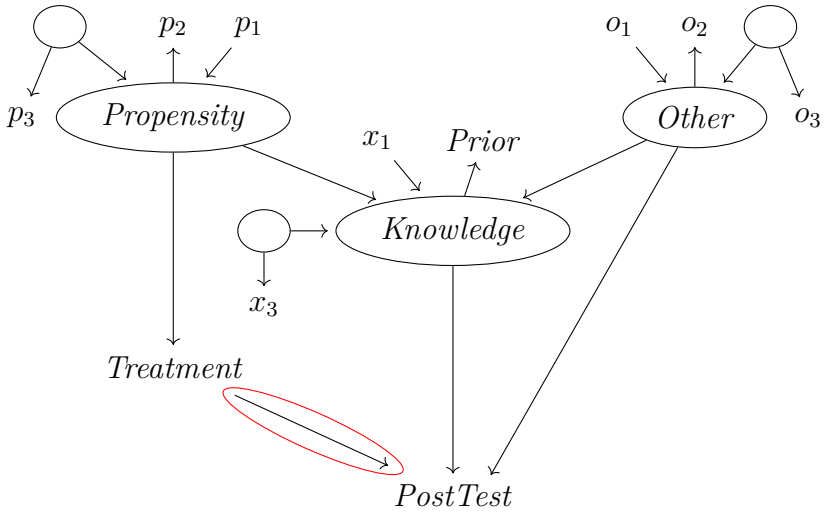

Figure 2. The data model with both latent and observed variables. The direct effect from *Treatment* to *PostTest*, enclosed in the red ellipse, is the construct the researchers wish to measure accurately. To make the figure less cluttered the individual node error variables are not shown.

The variable *Prior* is named in Figure 2 as opposed to just calling it $x2$ because of its central role in the gain score model. It is assumed that it is on the same scale as *PostTest* (in the simulation both have means of zero and standard deviations of one) so that *PostTest* − *Prior* is meaningful and that this difference represents the same *gain* throughout the span of *Prior*. *Prior* is influenced by *Knowledge*; it is a descendant of *Knowledge*. Therefore according to Pearl's second rule, conditioning on it unblocks the path *Treatment* ← *Propensity* → *Knowledge* ← *Other* → *PostTest*.

Analytic Strategies

Comparing scores on *PostTest* by *Treatment* using something like a t-test is not a good measure of the intervention when the two groups begin systematically different in ways associated with the outcome variable because differences on the outcome variable could be due to either the intervention or the pre-existing differences. Three procedures

will be considered in this paper: gain scores, Ancova, and propensity matching. Each of these is sometimes described as "controlling for previous performance."

The first two procedures are often discussed with reference to Lord's paradox (1967; 1969). Lord described a situation: looking at students' weights, before and after a year of college, where interest was with the gender difference. Lord imagined two statisticians proposing different methods for the analysis. The first statistician proposed subtracting the two weights and comparing means of these gain scores. The second statistician proposed predicting final weight from gender after conditioning by the initial weight using an Ancova. The statisticians came up with different results (the first statistician found no difference in weight gain, the second found males weighed more than females after conditioning on pre-weights). Several authors have shown when and why these approaches can produce different effects (e.g., Hand, 1994; Holland & Rubin, 1983; Pearl, 2016; Wainer, 1991; Wright, in press). Since Lord described this paradox, propensity matching (e.g. Rosenbaum, 2002; Rubin, 2006) has become very popular, though many express concern that it is being used without due concern (e.g., Pearl, 2009; Sekhon, 2009). Therefore it will also be compared.

Gain scores. The simplest of the procedures considered, computationally, is the gain score method. Let $Gain = PostTest - Prior$ and it is assumed that these scores have approximately the same meaning for each level of *Prior*. Analyses can then be conducted on this variable using *Treatment* with or without the other observed variables: $p_1 \ldots o_3$ (shown with the $\cdots$ in eqn. 1 below). Lord's (1967) first statistician conducted a t-test between the two groups on the gain score (i.e., no other covariates). One limitation of this approach is that it must make sense to equate, for example, Tom's increase from 96 to 99 with Jerry's increase from 47 to 50. In regression format this would be testing the hypothesis $\beta_1 = 0$ in:

$$Gain_i = PostTest_i - Prior_i = \beta_0 + \beta_1\, Treatment_i + \cdots + e_i \quad . \tag{1}$$

In R, where `Covs` is all covariates other than treatment and initial score, this would be:

```
lm(PostTest - Prior ~ Treatment + Covs)
```

Ancova. Lord's (1967) second statistician conducted an Ancova. This is the procedure most often described as *controling* covariates. Some people believe this, what Braun (2013) describes as magical thinking. There have been decades of warnings about the limitations of this procedure (e.g., Kahneman, 1965; Meehl, 1970). The phrase Ancova can mean different things to different people, but here it will refer to the model in eqn. 2:

$$PostTest_i = \beta_0 + \beta_1\, Treatment_i + \beta_2\, Prior_i + \cdots + e_i \quad . \tag{2}$$

This Ancova tests if $\beta_1 = 0$, like eqn. 1, but this β_1 is different. It is the effect after conditioning on *Prior* and the outcome variable is different (note that if β_2 is fixed at one this will fitted identically as eqn. 1). More covariates are often added to the regression, further obfuscating the meaning of β_1, sometimes with the optimistic hope that by including lots of variables this somehow gets closer to the isolating the influence of *Treatment* on *PostTest*. Including some covariates can make this more realistic, but including others will make it less realistic (e.g., conditioning on a collider in a backdoor path). Adding or removing a covariate changes the meaning of the parameter being estimated. The importance of thinking carefully about which covariates to include based on their role in the overall causal model and evaluating the performance of these statistical models with simulated data sets are the take-home messages of this paper. In R, where `Covs` is all covariates other than treatment, this would be:

```
lm(PostTest ~ Treatment + Prior + Covs)
```

Propensity Matching. Trying to reach causal conclusions when people are not randomly allocated into groups is difficult. Propensity matching attempts to create an accurate model of who chooses (or is chosen) to be in the intervention. If two people have

equal probabilities of being in the intervention, and if they differ in no other ways that are associated with the outcome and covariates, then which condition that they are in is effectively a flip of coin. The "differ in no other ways . . . " is an assumption that can be difficult to justify. Propensity matching was developed in a series of papers by Rosenbaum and Rubin. The seminal textbook is Rosenbaum (2002) and many of their contributions have been re-published in Rubin (2006). There are several statistics packages for performing propensity matching and estimating the treatment effects (see `www.biostat.jhsph.edu/~estuart/propensityscoresoftware.html`). R will be used for the simulation so Keller and Tipton's (2016) review of R propensity matching packages is particularly relevant.

Propensity matching can be divided into the following steps:

1. Choose a set of variables to estimate the probability of being in the treatment. This is the most important step and requires much content knowledge about how the variables are likely to relate.
2. Estimate the probability of being in the intervention, based on these covariates. Often this is done with a logistic regression, but there are many other possibilities.
3. Create matched samples of treatment and control participants. Several algorithms can be used for this, and particularly when the treatment is expensive/rare, it is common to match many control participants to each treatment participant.
4. Compare these matched samples to estimate the treatment effect. There are some options here, but for current purposes this will be the average treatment effect.

This can be implemented in R in many ways. Here `Covs` may or may not include `Prior`, depending on what the analyst believes. In the simulation the following is used:

```
propval <- glm(Treatment ~ Covs,family=binomial)$fitted
library(Matching)
Match(PostTest,Treatment,propval,
      estimand="ATE",BiasAdjust=TRUE,ties=TRUE)
```

`Match` is a function from the **Matching** package (Sekhon, 2011) and the code on the second line of this function call are specific options for this function. The only of note is `estimand="ATE"`, which means the estimate is the average treatment effect. The user also has the option to estimate the treatment effect for those in the treatment or for those in the control group. These would be valuable for different applied problems. The former to estimate the effect for those likely to sign up for the treatment and the later the value if you could encourage those who would not normally sign up to sign up.

There are differences between propensity matching and Ancova (e.g., that matching is used for conditioning in one). The most important difference for the current paper is that with Ancova the covariates are used to predict the outcome (so models might maximize R^2 for the final test scores) while with propensity matching the covariates are used to predict being in the treatment condition. The goal for propensity is to have matched cases with equal probability of being in the treatment group. If it can be argued that there are two people each with 62% chance of being in the math club, and no other differences, then it can be argued that this is like an experiment. In practice, there are likely other differences among those in the math clubs than those not. Rosenbaum (2002) describes methods to examine this and many of these are implemented in the available software, but there may be differences that are not captured by any of the observed variables.

Simulation Methods

Simulation methods are used here to explore the choice of statistical procedures and which covariates to include. These methods allow data for causal models to be created several thousand times and different statistical methods applied and evaluated. They also allow the causal models to be systematically manipulated to examine further research questions. For example, the path *Other* $\rightarrow$ *PostTest* can be removed (which would mean *Knowledge* is no longer a collider in a backdoor path, and this is considered in the first extension below) and different statistical models evaluated. The R statistics environment

(R Core Team, 2018, version 3.6.1) is used. Other environments/packages (e.g., Python, SAS, SPSS) could have been used, but using R allows use of its propensity matching packages (e.g., Keller & Tipton, 2016) and it is freely available to all readers.

Eleven sets of covariates were used for each procedure. These were: a model without any covariates (except for the propensity matching procedures, since the propensities would all be the same so the matching algorithm will not work), each of the nine covariates ($p1 \ldots o3$) on its own (except using *Prior* as a covariate with the gain score model since, as noted above, this would be equivalent to the Ancova), and all of the covariates. The gain score and Ancova models both used R's `lm` function. Propensity matching used R's `glm` function (logistic regression) to calculate propensity scores and the `Match` function (Sekhon, 2011) to estimate the average treatment effect.

The main simulation study and the extensions had $n = 200$ and there were 10,000 replications per condition. This number of replications was used so that the standard error for the z value for each statistical model was approximately 0.01. The three observed variables associated with the latent variables were constructed to have population correlations of $r = .50$ with their latent variable. The contributions from *Propensity* and *Other* to *Knowledge* are equal, and a normally distributed random error is also added. *PostTest* is based on equal contributions from *Other* and *Knowledge* (and random error). There is no influence from *Treatment* to *PostTest*. Non-zero effects could be included to evaluate the power of different methods and to explore the association between the estimated and true effects. The variables (other than *Treatment*) are all scaled to have a mean of zero and a standard deviation of one. The code to create the data for this study are shown in the appendix.

Simulation Results

Table 1 gives the mean of the test statistic (t or z) for the 10,000 replications for each statistical procedure for each of the eleven sets of covariates. Values above zero correspond

	Gain	Ancova	Propen
none	-0.02	-1.74	
p1	-0.02	-1.36	-1.17
p2	-0.02	-1.36	-1.18
p3	-0.02	-1.36	-1.17
x1	-0.02	-1.81	-1.56
Prior		-1.20	-1.03
x3	-0.02	-1.80	-1.56
o1	-0.03	-1.88	-1.63
o2	-0.02	-1.87	-1.61
o3	-0.02	-1.87	-1.62
all	-0.02	-0.97	-0.60
Mean	-0.02	-1.57	-1.31

Table 1
The mean of the test statistic (t or z) of the 10,000 replications for each statistical procedure for each covariate. The standard errors of these are approximately 0.01.

to mean values suggesting that the treatment has a positive effect and values below zero suggest the treatment has a negative effect. Because these are simulated data it is known that the correct answer is that the treatment had no effect.

The result that stands out is that the gain score method appears good. The mean test statistics are only slightly lower than the null. The propensity matching procedure does slightly better (i.e., means closer to zero) for all choices of covariates than the Ancova procedure, but each are biased downwards for the data model in Figure 2. These tend to estimate a negative treatment effect. For the Ancova and propensity matching procedures, conditioning on *Prior* does better than the other covariates. Of the covariate sets, those related to *Propensity* reduces the bias the most, and then those with *Knowledge*, and finally those with *Other*.

A single simulation can show that the statistical procedures can work differently, but particularly when the model includes relationships involving latent variables—where the values used to create the data are likely speculative—it is worth checking how sensitive any conclusions are to variations in the causal model. Thus, the conclusion "the gain score method appears good" may be premature.

Some Extensions

Many assumptions are made when constructing simulated data. A valuable attribute of simulations is that if people question any of the assumptions or want to test how varying different aspects affect the results, this can be done. For illustrative purposes two extensions are shown here.

Varying the Strength of the Backdoor Paths. The first extension varies the strength of the two backdoor paths. In the first simulation it was assumed that *PostTest* was equally influenced by *Knowledge*, *Other*, and random error, but these contributions will vary by situation so it is important to determine if any conclusions are sensitive to the relative contributions of these. Further, the two edges, *Knowledge* $\rightarrow$ *PostTest* and *Other* $\rightarrow$ *PostTest*, are part of different backdoor paths. *Knowledge* $\rightarrow$ *PostTest* is part of a path that begins unblocked, but becomes partially blocked when conditioning on covariates related to *Knowledge*. *Other* $\rightarrow$ *PostTest* is part of a path that begins blocked because *Knowledge* is a collider in this path, but is unblocked when conditioning on *Prior* because it is a descendant of *Knowledge*.

For this extension the *PostTest* variable will be determined by:

$$X\ Knowledge + (2 - X)\ Other + \mathcal{N}(0, 1) \quad ,$$

where X varies from 0–2 so that the mid-point of this sequence is the same as with the initial simulation. When $X = 0$ this means that *PostTest* is influenced by *Other*, but not *Knowledge*. When $X = 2$ *PostTest* is influenced by *Knowledge*, but not *Other*. For this simulation X is allowed to vary in 11 steps from 0 to +2, and just the model with all covariates included is reported.

Figure 3 shows that the estimated treatment effect is higher, for all values of X, for the gain score method, followed by the propensity score methods, and then the Ancova. All underestimate the treatment effect when the effect of *Knowledge* is greater than the effect

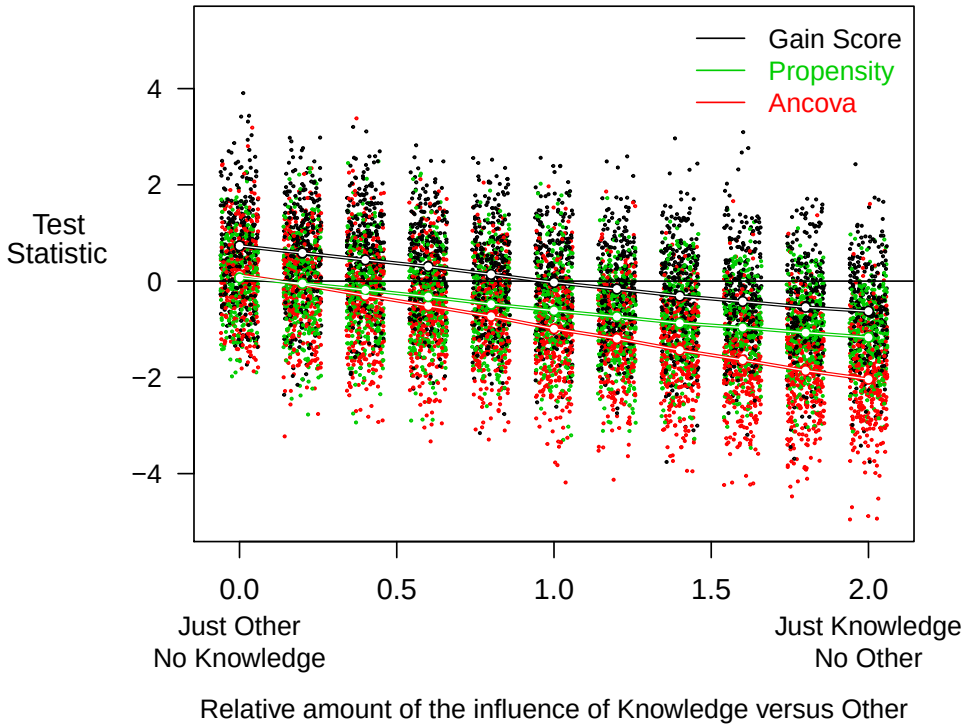

Figure 3. Test statistic values by statistical procedure with the the relative impact of *Other* and *Knowledge* on *PostTest*. Note: Only 10,000 of the points are plotted. A random *jitter* has been added to x-axis values.

of *Other* (the right side of the plot, $X > 1$). When the relative effects of these two are the same ($X = 1$), as with the above simulation, the gain score method correctly finds no treatment effects. When there is no *Knowledge* influence on *Propensity* ($X = 0$), the propensity matching and Ancova methods estimate almost no treatment effect, and the gain score method incorrectly estimates a positive treatment effect. This shows that it is important to explore different causal models to decide which may be more appropriate. If there is uncertainty, for example, in the relative impact of these two effects, it is worth reporting this sensitivity, report any empirical evidence about which causal models appear more plausible, and in some cases report multiple analyses. Conclusions should be tentative if different procedures, chosen because of their fit for plausible causal models, lead to different conclusions.

Having the Covariate Influence the Treatment. The second extension focuses on whether the covariate has a causal impact on treatment condition. In some situations an assessment is used both to assign people to groups (e.g., a remedial or accelerate program in education) and as a baseline measure. There are concerns using a covariate in this way, but it occurs. Suppose that *Prior* influences the propensity to be assigned to the treatment condition for the graph in Figure 2. All else being equal, people with high *Prior* scores will likely regress down a bit towards the sample mean on their *PostTest*, and the opposite for people with low scores (Galton, 1886). Suppose that high *Prior* scores influences whether someone is in the treatment condition. In this circumstance you would expect many of those in the treatment to regress down towards the sample mean and many of those not in the treatment condition to regress up towards the sample mean. Thus, the gain score procedure will often show that the treatment is detrimental even when the true treatment effect is zero. Suppose *Prior* scores are associated, but not directly causally related to treatment allocation. Both might be caused by a third variable, like really liking math. For explanatory purposes, suppose there are two groups: math lovers and math non-lovers. Now on the *PostTest* you would expect people to regress towards their groups' mean (which will likely be higher for the math lovers group). If you compare two people with similar *Prior* scores from these two groups (as done with Ancova and matching procedures), you would expect the person in the math lovers group to have the higher *PostTest* score because that person is regressing towards the higher mean than the person in the math non-loving group. Since there will be more math lovers in the treatment condition, you would expect the Ancova and matching procedures to show a positive treatment effects even when the true treatment effect is zero.

When building extensions to evaluate statistical procedures for causal models it can be useful to add edges to the graphs, but it is also often useful to simplify the graph to isolate why the differences occur. A series of simple models are depicted in Figure 4. In Panel A both *Knowledge* $\rightarrow$ *Propensity* and *Prior* $\rightarrow$ *Propensity* exist. Here, the gain score

procedure estimates a positive treatment effect when the true treatment effect is zero, but the other procedures estimate a negative treatment effect (Table 2). They are biased in different directions, and the results for the next two panels show why. In Panel B *Prior* no longer affects *Treatment*, and the gain score procedure provides unbiased estimates of the treatment effect but the Ancova and matching procedures estimate a negative treatment effect. Panel C has only *Prior* (and random variation) influencing *Propensity*. Now the Ancova and propensity matching procedures provide unbiased estimates, and the gain score procedure incorrectly estimates a positive treatment effect.

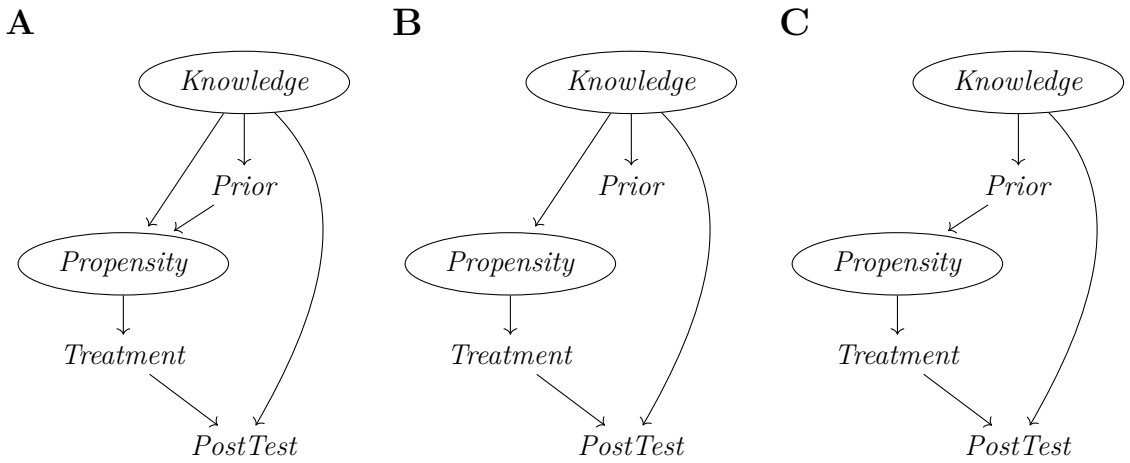

Figure 4. Three ways in which *Knowledge* and *Prior* may influence *Propensity*. In Panel A both influence it, in Panel B only *Knowledge* influences it, and in Panel C only *Prior* influences it.

Conclusion

Gain score, Ancova, and propensity matching are all used with the hope of isolating the direct effect of an intervention on an outcome measure. No statistical procedure can guarantee to isolate this direct effect in all situations, but each can be useful depending on the causal model that led to the data. None of these–or any statistical procedure–will work in all situations. When more effects are added to the causal/data model it can become impossible to block all backdoor paths with any statistical procedure. There are situations

Table 2
The mean test statistic for the three statistical procedures for each panel in Figure 4.

		Statistical Procedure		
		Gain	Ancova	Prop. Matching
Panel	A	1.90	-1.74	-1.51
	B	0.01	-2.41	-2.09
	C	2.86	0.02	-0.04

where no statistical procedure can accurately estimate a treatment effect. The goal is to choose which procedure is most accurate and to warn readers about this limitation.

The results here show that there are situations where the gain score model does better than the Ancova and propensity matching procedures, and situations where it does worse. The propensity matching procedures tended to outperform Ancova methods here. The choice of covariates is important. Choosing those associated with propensity performed relatively well, as did using *Prior*. The main conclusion, however, is that the choice of statistical procedure depends on assumptions about how the data arose.

Unfortunately the analyst will not know the true causal model underlying the data. Assumptions must be made in order to make causal conclusions (Cartwright, 2014). The analyst can try several causal models, as was done with the extensions here, to test if the statistical approach is sensitive to these changes. The extensions here were specifically chosen to show that the procedures can be sensitive to changes, but not all changes have these effects. Researchers should be prepared for reviewers and readers to propose their own causal models. Enough information should be provided to allow these peers to simulate data to show if the statistical procedure used would be appropriate for their choice of causal model. When there are differences, it is often necessary to conduct further research designed to evaluate which set of causal models is more appropriate.

This can be time-consuming. Statistical procedures for intervention studies without random allocation are not simple, hence this special issue. Plotting the data and exploratory/descriptive data analysis may be useful, though the choice of these is often

influenced by the assumptions made. If you are uncertain which procedure to use, you can use multiple approaches, but must describe this in the write-up (i.e., not use each and just report the one with the lowest p-value). Steegen et al. (2016) take this to an extreme advising analysts to run many potential models and then report the distribution of results and weight them for plausibility. Here, the advice is to try to limit the choice to a small number of plausible models.

References

Braun, H. I. (2013). Value-added modeling and the power of magical thinking. *Ensaio: Evaluation of Public Policies in Education [Brazil]*, *21*, 115–130. doi: 10.1590/S0104-40362013000100007

Cartwright, N. (2014). Causal inference. In N. Cartwright & E. Montuschi (Eds.), *Philosophy of social science* (pp. 308–326). Oxford, UK: Oxford University Press.

Elwart, F. (2013). Causal graphical models. In S. L. Morgan (Ed.), *Handbook of causal analysis for social research* (pp. 245–273). New York: Sage Publications.

Galton, F. (1886). Regression towards mediocrity in hereditary stature. *The Journal of the Anthropological Institute of Great Britain and Ireland*, *15*, 246–263. Retrieved from `www.jstor.org/stable/2841583`

Hand, D. J. (1994). Deconstructing statistical questions. *Journal of the Royal Statistical Society. Series A (Statistics in Society)*, *157*, 317–356. doi: 10.2307/2983526

Hill, C., Abraham, C., & Wright, D. B. (2007). Can theory-based messages in combination with cognitive prompts promote exercise in classroom settings? *Social Science & Medicine*, *65*, 1049–1058. doi: https://doi.org/10.1016/j.socscimed.2007.04.024

Holland, P. W., & Rubin, D. B. (1983). On Lord's paradox. In H. Wainer & S. Messick (Eds.), *Principals of modern psychological measurement* (pp. 3–35). Hillsdale, NJ: Erlbaum.

Imbens, G. W., & Rubin, D. B. (2015). *Causal inference for statistics, social, and biomedical sciences: An introduction.* New York, NY: Cambridge University Press.

Kahneman, D. (1965). Control of spurious association and the reliability of the controlled variable. *Psychological Bulletin*, *64*, 326–329.

Keller, B., & Tipton, E. (2016). Propensity score analysis in R: A software review. *Journal of Educational and Behavioral Statistics*, *41*, 326–348. doi: 10.3102/1076998616631744

Loehlin, J. C. (1998). *Latent variable models: An introduction to factor, path, and structural analysis.* (Third ed.). Mahwah, NJ: Erlbaum.

Lord, F. M. (1967). A paradox in the interpretation of group comparisons. *Psychological Bulletin*, *68*, 304–305. doi: 10.1037.h0025105

Lord, F. M. (1969). Statistical adjustments when comparing preexisting groups. *Psychological Bulletin*, *72*, 336–337. doi: 10.1037/h0028108

Meehl, P. E. (1970). Nuisance variales and the ex post facto design. In M. Radner & S. Winokur (Eds.), *Minnesota studies in the philosophy of science: Vol IV. Analysis of theories and methods of physics and psychology* (pp. 373–402). Minneapolis, MN: University of Minnesota Press.

Morgan, S. L., & Winship, C. (2007). *Counterfactuals and causal inference: Methods and principles for social research*. Cambridge, MA: Cambridge University Press.

Pearl, J. (2009). *Causality: Models, reasoning, and inference (2nd ed.)*. New York: Cambridge University Press.

Pearl, J. (2016). Lord's paradox revisited (Oh Lord! Kumbaya!). *Journal of Causal Inference*, *4*(2). doi: 10.1515/jci-2016-0021

Pearl, J., Glymour, M., & Jewell, N. P. (2016). *Causal inference in statistics: A primer*. Chichester, UK: Wiley.

Peters, J. (2013, July). *When ice cream sales rise, so do homicides. coincidence, or will your next cone murder you?* Slate. Retrieved from `slate.com/news-and-politics/2013/07/warm-weather-homicide-rates-when-ice-cream-sales-rise-homicides-rise-coincidence.html`

R Core Team. (2018). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `https://www.R-project.org/`

Rosenbaum, P. R. (2002). *Observational studies* (2nd ed.). New York, NY: Springer-Verlag. doi: 10.1007/978-1-4757-3692-2

Rubin, D. B. (2006). *Matched sampling for causal effects*. Cambridge, MA: Cambridge University Press. doi: 10.1017/CBO9780511810725

Sekhon, J. S. (2009). Opiates for the matches: Matching methods for causal inference.

Annual Review of Political Science, *12*, 487–508.

Sekhon, J. S. (2011). Multivariate and propensity score matching software with automated balance optimization: The **Matching** package for R. *Journal of Statistical Software*, *42*(7), 1–52. Retrieved from `http://www.jstatsoft.org/v42/i07/`

Steegen, S., Tuerlinckx, F., Gelman, A., & Vanpaemel, W. (2016). Increasing transparency through a multiverse analysis. *Perspectives on Psychological Science*, *11*, 702–712.

Wainer, H. (1991). Adjusting for differential base rates: Lord's paradox again. *Psychological Bulletin*, *109*, 147–151.

Wright, D. B. (2017). Using graphical models to examine value-added models. *Statistics and Public Policy*, *4*, 1–7. doi: 10.1080/2330443X.2017.1294037

Wright, D. B. (2018). Estimating school effectiveness with student growth percentile and gain score models. *Journal of Applied Statistics*, *45*, 2536-2547. doi: 10.1080/02664763.2018.1426742

Wright, D. B. (in press). Allocation to groups: Examples of Lord's paradox. *British Journal of Educational Psychology*.

Appendix

R Code for the simulations

The code to create the data used in the main simulation is:

```
n <- 200
reps <- 10000 #47
set.seed <- 49811
vals <- matrix(ncol=6,nrow=reps*(12*3))
for (i in 1:reps){
  pph <- rnorm(n)
  p1 <- rnorm(n)
  p3 <- scale(pph + 1.0371*rnorm(n)) #
  propen <- scale(.3*pph + .2082*p1  + .2*rnorm(n))
  p2 <- scale(sqrt(.25)*propen + sqrt(.75)*rnorm(n))
  oph <- rnorm(n)
  o1 <- rnorm(n)
  o3 <- scale(oph + 1.0371*rnorm(n)) #
  other <- scale(.3*oph + .2082*o1 + .2*rnorm(n))
  o2 <- scale(sqrt(.25)*other + sqrt(.75)*rnorm(n))
  xph <- rnorm(n)
  x1 <- rnorm(n)
  x3 <- scale(xph + .71*rnorm(n)) #
  know <- scale(.85*xph + .6935*x1 + .6*other + .6*propen)
  prior <- scale(sqrt(.25)*know + sqrt(.75)*rnorm(n))
  propenval <- qrank(propen)
  treat <- rbinom(n,1,propenval)
  post <- scale(other + 0*treat + know + rnorm(n))
  vals[((i-1)*36+1):(i*36),] <- allmodels(p1,p2,p3,x1,prior,x3,
                                    o1,o2,o3,treat,post,know)
    }

colnames(vals) <-
    c("model","statmodel","ATE","seATE","zval","pval")
vals <- as.data.frame(vals)
vals$statmodel <-
  factor(recoder(vals$statmodel,'1:"none";2:"p1";3:"p2";4:"p3";
                 5:"x1";6:"Prior";7:"x3";8:"o1";9:"o2";10:"o3";
                 11:"all";12:"Knowledge" '),
         levels=c("none","p1","p2","p3","x1","Prior","x3",
                  "o1","o2","o3","all","Knowledge"))
vals$model <-
```

```
  factor(recoder(vals$model,'0:"Gain";1:"Ancova";2:"Propen" '),
         levels=c("Gain","Ancova","Propen"))
```

The function `allmodels` runs all the statistical models and returns a six-column matrix of: the procedure (gain score, Ancova, propensity matching), which covariates are included, the effect estimate, its standard error, the t (from `lm`) or z (from `glm` and `Match`), and the associated two-tailed p-value. It calls three functions, `gain`, `anc`, and `pm`, which run the models for the gain score, Ancova, and propensity matching procedures, respectively. The package **Matching** (Sekhon, 2011) needs to be installed and loaded since it is called by the `pm` function. This function and the functions it calls are here:

```
# 0 gain, 1 ancova, 2 pm
gain <- function(p1,p2,p3,x1,prior,x3,o1,o2,o3,treat,post,know){
     gainvals <- matrix(ncol=6,nrow=12)
     gainvals[1,] <- c(0,1,summary(lm(post - prior ~ treat))$coef[2,])
     gainvals[2,] <- c(0,2,summary(lm(post - prior ~ treat + p1))$coef[2,])
     gainvals[3,] <- c(0,3,summary(lm(post - prior ~ treat + p2))$coef[2,])
     gainvals[4,] <- c(0,4,summary(lm(post - prior ~ treat + p3))$coef[2,])
     gainvals[5,] <- c(0,5,summary(lm(post - prior ~ treat + x1))$coef[2,])
     gainvals[6,] <- c(0,6,NA,NA,NA,NA) #since same as Ancova
     gainvals[7,] <- c(0,7,summary(lm(post - prior ~ treat + x3))$coef[2,])
     gainvals[8,] <- c(0,8,summary(lm(post - prior ~ treat + o1))$coef[2,])
     gainvals[9,] <- c(0,9,summary(lm(post - prior ~ treat + o2))$coef[2,])
     gainvals[10,] <- c(0,10,summary(lm(post - prior ~ treat +
                                        o3))$coef[2,])
     #note prior below
     gainvals[11,] <- c(0,11,summary(lm(post - prior ~ treat +
                    p1 + p2 + p3 + x1 + x3 + o1 + o2 + o3))$coef[2,])
     gainvals[12,] <- c(0,12,summary(lm(post - prior ~ treat +
                                        know))$coef[2,])
        return(gainvals) }

anc <- function(p1,p2,p3,x1,prior,x3,o1,o2,o3,treat,post,know){
     ancvals <- matrix(ncol=6,nrow=12)
     ancvals[1,] <- c(1,1,summary(lm(post ~ treat))$coef[2,])
     ancvals[2,] <- c(1,2,summary(lm(post ~ treat + p1))$coef[2,])
     ancvals[3,] <- c(1,3,summary(lm(post ~ treat + p2))$coef[2,])
     ancvals[4,] <- c(1,4,summary(lm(post ~ treat + p3))$coef[2,])
     ancvals[5,] <- c(1,5,summary(lm(post ~ treat + x1))$coef[2,])
```

```
     ancvals[6,] <- c(1,6,summary(lm(post ~ treat + prior))$coef[2,])
     ancvals[7,] <- c(1,7,summary(lm(post ~ treat + x3))$coef[2,])
     ancvals[8,] <- c(1,8,summary(lm(post ~ treat + o1))$coef[2,])
     ancvals[9,] <- c(1,9,summary(lm(post ~ treat + o2))$coef[2,])
     ancvals[10,] <- c(1,10,summary(lm(post ~ treat + o3))$coef[2,])
     ancvals[11,] <- c(1,11,summary(lm(post ~ treat +
          p1 + p2 + p3 + x1 + prior + x3 + o1 + o2 + o3))$coef[2,])
     ancvals[12,] <- c(1,12,summary(lm(post ~ treat + know))$coef[2,])
          return(ancvals) }


pm <- function(p1,p2,p3,x1,prior,x3,o1,o2,o3,treat,post,know){
       pmvals <- matrix(ncol=6,nrow=12)
       pmvals[1,] <- c(2,1,NA,NA,NA,NA)
       propval <- glm(treat~ p1,family="binomial")$fitted
       mod <-   Match(post,treat,propval,estimand="ATE",
                      BiasAdjust=TRUE,ties=TRUE)
       pmvals[2,] <- c(2,2,mod$est,mod$se,
                       zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

       propval <- glm(treat~ p2,family="binomial")$fitted
       mod <-   Match(post,treat,propval,estimand="ATE",
                      BiasAdjust=TRUE,ties=TRUE)
       pmvals[3,] <- c(2,3,mod$est,mod$se,
                       zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

       propval <- glm(treat~ p3,family="binomial")$fitted
       mod <-   Match(post,treat,propval,estimand="ATE",
                      BiasAdjust=TRUE,ties=TRUE)
       pmvals[4,] <- c(2,4,mod$est,mod$se,
                       zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

       propval <- glm(treat~ x1,family="binomial")$fitted
       mod <-   Match(post,treat,propval,estimand="ATE",
                      BiasAdjust=TRUE,ties=TRUE)
       pmvals[5,] <- c(2,5,mod$est,mod$se,
                       zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

       propval <- glm(treat~ prior,family="binomial")$fitted
       mod <-   Match(post,treat,propval,estimand="ATE",
                      BiasAdjust=TRUE,ties=TRUE)
       pmvals[6,] <- c(2,6,mod$est,mod$se,
                       zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)
```

```
      propval <- glm(treat~ x3,family="binomial")$fitted
      mod <-   Match(post,treat,propval,estimand="ATE",
                     BiasAdjust=TRUE,ties=TRUE)
      pmvals[7,] <- c(2,7,mod$est,mod$se,
                      zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

      propval <- glm(treat~ o1,family="binomial")$fitted
      mod <-   Match(post,treat,propval,estimand="ATE",
                     BiasAdjust=TRUE,ties=TRUE)
      pmvals[8,] <- c(2,8,mod$est,mod$se,
                      zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

      propval <- glm(treat~ o2,family="binomial")$fitted
      mod <-   Match(post,treat,propval,estimand="ATE",
                     BiasAdjust=TRUE,ties=TRUE)
      pmvals[9,] <- c(2,9,mod$est,mod$se,
                      zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

      propval <- glm(treat~ o3,family="binomial")$fitted
      mod <-   Match(post,treat,propval,estimand="ATE",
                     BiasAdjust=TRUE,ties=TRUE)
      pmvals[10,] <- c(2,10,mod$est,mod$se,
                      zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

      propval <- glm(treat~ p1 + p2 + p3 + x1 + prior +
                       x3 + o1 + o2 + o3,family="binomial")$fitted
      mod <-   Match(post,treat,propval,estimand="ATE",
                     BiasAdjust=TRUE,ties=TRUE)
      pmvals[11,] <- c(2,11,mod$est,mod$se,
                      zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

         propval <- glm(treat~ know,family="binomial")$fitted
      mod <-   Match(post,treat,propval,estimand="ATE",
                     BiasAdjust=TRUE,ties=TRUE)
      pmvals[12,] <- c(2,12,mod$est,mod$se,
                      zval<-mod$est/mod$se,(1 - pnorm(abs(zval)))*2)

      return(pmvals)
      }

allmodels <- function(p1,p2,p3,x1,prior,x3,o1,o2,o3,treat,post,know){
  dmvals <- matrix(ncol=6,nrow=(12*3) )
  dmvals[1:12,] <- gain(p1,p2,p3,x1,prior,x3,o1,o2,o3,treat,post,know)
  dmvals[13:24,] <- anc(p1,p2,p3,x1,prior,x3,o1,o2,o3,treat,post,know)
```

```
  dmvals[25:36,] <- pm(p1,p2,p3,x1,prior,x3,o1,o2,o3,treat,post,know)
  return(dmvals)
}

qrank <- function(x) 1-(rank(x)+1)/(length(x)+2)
```

The code for these, the extensions, and the entire paper are also available at: `https://github.com/MASKED`. Readers are encouraged to adapt the code for their own needs.