Abstract: Instrumental variables estimators are known to be biased but consistent. This research note simulates the bias associated with instrumental variables estimators of experimental treatment effects. We find the bias to be negligible in size, except for extreme cases in which the contact rates and sample sizes are very small.
This paper responds to Imai (2002, 2003), who emphasizes the fact that ratio estimators such as the instrumental variable estimator discussed by Angrist et al. (1990) are biased. It is, of course, a well-known fact that the instrumental variables estimator is biased (cf. Wooldridge, 2000, p.465). This research note points out, however, that the magnitude of this bias is negligible, except in extreme cases.

This point may be demonstrated by means of a Monte Carlo simulation. For concreteness, we consider the case described by Gerber and Green (2000) in which the dependent variable is whether a person votes. The experimental treatment is whether a voter is encouraged to vote prior to the election, but only some of those who are assigned to the treatment group are in fact contacted by canvassers. Thus, the population of eligible voters may be divided into two groups, those who are reachable by canvassers and those who are not reachable.

The probability of voting among those assigned to the treatment group may be expressed as a weighted average of the voting rates in the reachable ($p_r$) and nonreachable ($p_{nr}$) groups

\begin{equation}
V_e = \alpha (p_r +t) + (1-\alpha) p_{nr},
\end{equation}

where $\alpha$ represents the proportion of eligible voters who are reachable.

The probability of voting among those randomly assigned to the control group may be assumed to reflect the same underlying proportions of reachable and nonreachable people, but without a treatment effect:

\begin{equation}
V_c = \alpha p_r + (1-\alpha) p_{nr}.
\end{equation}
The parameter $\alpha$ may be estimated using the proportion of people assigned to the treatment group who are actually reached. Let us call this estimator $\alpha^*$. Manipulating equations (1) and (2) provides an estimator ($t^*$) for the treatment effect $t$:

$$(3) \quad t^* = (V_e - V_c)/\alpha^*.$$  

Because the denominator of this expression is a random variable, the expectation of $t^*$ will not equal $t$ in small samples. The question is whether this bias is sufficiently large to be of practical concern.

Our Monte Carlo simulation examines the size of the bias when $t=.10$, $p_r=.55$, $p_{nr} = .40$, which are approximately the values obtained by Gerber and Green (2000). Their contact rate was 29%, but we consider here contact rates of .05, .25, .5, and .75. We also consider sample sizes of 100, 1000, and 10000, where sample size is defined as the number of subjects assigned to the treatment group (with like numbers assigned to the control group). For each pair of contact rates and sample sizes, 100,000 samples were generated. Table 1 reports the means and standard deviations in each resampling experiment.

The conclusion to be drawn from this exercise is that the biases are trivial in size. For samples of 1000 or more (which would include the Gerber and Green 2000 study), the biases are negligible. The fact that they are slightly negative reflects the fact that the sampling distributions of $\alpha^*$ and $(V_e - V_c)$ are slightly positively correlated in this example – when the treatment group contains more that the expected number of reachable voters, the intent-to-treat effect represented by $(V_e - V_c)$ tends to be greater. The only instance in which the biases are serious occurs when the contact rate is very low (.05) and the sample size is very small (100). This case would seldom arise in practice given the very low power of such an experiment. When the contact rate is .25, the bias becomes negligible even for $n=100$.  

2
References


Table 1: Monte Carlo Results for .10 Treatment Effect, for Varying Sample Sizes and Contact Rates (100,000 Replications)

<table>
<thead>
<tr>
<th>Sample Size</th>
<th>Contact Rate</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>.05</td>
<td>.10</td>
<td>.25</td>
<td>.50</td>
<td>.75</td>
</tr>
<tr>
<td>100</td>
<td>.053*</td>
<td>.081</td>
<td>.098</td>
<td>.099</td>
<td>.100</td>
</tr>
<tr>
<td></td>
<td>(2.126)</td>
<td>(.847)</td>
<td>(.296)</td>
<td>(.144)</td>
<td>(.094)</td>
</tr>
<tr>
<td>500</td>
<td>.094</td>
<td>.098</td>
<td>.099</td>
<td>.100</td>
<td>.100</td>
</tr>
<tr>
<td></td>
<td>(.663)</td>
<td>(.323)</td>
<td>(.128)</td>
<td>(.064)</td>
<td>(.042)</td>
</tr>
<tr>
<td>1000</td>
<td>.098</td>
<td>.099</td>
<td>.100</td>
<td>.100</td>
<td>.100</td>
</tr>
<tr>
<td></td>
<td>(.456)</td>
<td>(.224)</td>
<td>(.090)</td>
<td>(.045)</td>
<td>(.030)</td>
</tr>
<tr>
<td>10000</td>
<td>.100</td>
<td>.099</td>
<td>.100</td>
<td>.100</td>
<td>.100</td>
</tr>
<tr>
<td></td>
<td>(.139)</td>
<td>(.070)</td>
<td>(.028)</td>
<td>(.014)</td>
<td>(.009)</td>
</tr>
</tbody>
</table>

*Based on 99,390 samples in which the contact rate was greater than 0.
Appendix: Gauss Code Used for Simulation

@ program to simulate the effects of estimating alpha @
@ simulates draws of data of size n @

n=100;      @ number of observations in EACH group (treatment/control) @
alpha=.25;     @ specify the average contact rate @
p_nr=.4;      @ intercept for nonreachable population @
p_r=.55;      @ intercept for reachable population @
t=.1;         @ treatment effect @

let b_stack[1,5] = . . . . . ;
repls=100000;  @ number of replications in bootstrap @
j=0;

do while j < repls;

alphahat=meanc(floor(rndu(n,1)+alpha));

v_e=meanc(floor(rndu(n,1)+alphahat*(p_r+t)+(1-alphahat)*p_nr));
v_c=meanc(floor(rndu(n,1)+alpha*(p_r)+(1-alpha)*p_nr));

b_iv=(v_e-v_c)/alphahat;  @ instrumental variables estimator @
bstats=alpha~alphahat~v_e~v_c~b_iv;
b_stack=b_stack|bstats;
j=j+1;
endo;

e=(b_stack[..,2] .gt 0);
b_stack=selif(b_stack,e);

print "means of alpha~alphahat~v_e~v_c~b_iv " ;
meanc(b_stack)';
stdc(b_stack)' " <------ standard errors";
print "based on " repls " replications, of which " rows(b_stack) " have alphahat > 0";

corr_b=corrx(b_stack[..,2]~(b_stack[..,3]-b_stack[..,4]));
print corr_b[2,1] " correlation between b_iv and alphahat for n=" n;
print "------------------------------------------";