Introduction

For estimating causal relations from surveys with missing data, Eff and Dow’s (2009) revolutionary solution published in this journal marks a milestone in social science, one carefully worked out and proven in the field of inferential statistics.¹ For anthropology and cross-cultural surveys, such as regression analysis of data from the Standard Cross-Cultural Sample (SCCS) collaborative project,² which is freely accessible for researchers and classroom use, UC Irvine’s undergraduates are the first to experiment with use the approach. Here I provide a roadmap to help instructors in the classroom and help students benefit from our experiences in the classroom.

Digital Learning

Digital media, such as the Mediawiki software used by Wikipedia, are the key to classroom learning solutions involving complex open-source software that can be installed for free in a lab and simultaneously in students’ abodes. In our approach each student begins with their own wiki page into which the program is pasted into an edit window and then pasted into R to run. Program results are posted in a separate edit window, and successive program modifications (“EduMods”) and new results are placed in successive pairs of labeled edit windows. Thus pairs of edit windows are successively added to a single wiki page until the final results are completed and the student can write up their findings for the research questions they have posed. The authors and general editor for the programs worked to make it easy to use the R programs which do the regression analysis of the 186-society SCCS data.

R code for SCCS (the Standard Cross-Cultural Sample database)

The R code is divided into two parts: **Program 1** reads the SCCS data³ and imputes missing data, and **Program 2** defines the dependent variable (depvar) and does two-stages of regression (2SLS: two-stage least squares), the first for the effects of clusters of societies that are similar on the depvar in terms of those with (1) closer distances and (2) language similarities and the second for causal predictors of the depvar. The procedure for each student is to:

1. Copy the program and data from the eJournal source files onto the student home and class computers.

---

² The SCCS dataset in R was one that James Dow (2004) had improved upon from the SPSS version by Khaltourina, Korotayev and Divale (2002) which drew on corrections from Gray (1999) and earlier versions by White and Burton.
³ The program can be changed to read and analyze other survey data from any field of study.
2. Unzip the files on a directory specially named for the classroom computer lab (at UCI our lab uses setwd("C:/My Documents/MI"), where setwd is the R command to set the working directory. Copy the lines at the top of the program where the SCCS.Rdata and Vaux.Rdata files are read. Vaux is for the auxiliary Variables file as described in Eff and Dow (2009). Use the slider to move up and down the output of R to find any errors that involve reading of data, and correct the errors. Then copy, paste, run and correct the code down to include the line in the first box below (this includes the naming and definition of the independent variables from the SCCS$ dataset). Then do the same for the code that includes the starting and ending lines in the following box (this includes missing values and descriptive statistics for the independent variables).

```r
#--look at first 6 rows of fx--
head(fx)  # Box 1

#------MODIFICATIONS BELOW THIS LINE--------
#--check to see number of missing values--
... (the code here will collect statistics on the independent variables)
#------MODIFICATIONS ABOVE THIS LINE--------  # Box 2
```

3. Now copy and paste the remainder of the program, check for errors, and if none occur, save these Eff and Dow results for the Restricted model.

**Edit Windows in the Wiki**

Students work within a wiki (at UCI, hosted at http://intersci.ss.uci.edu/wiki), copying and pasting into R and similarly from the results in R back to the wiki. The student EduMod pages for this experiment are found from the wiki search window, using lab001 as the search request (distinctive keywords help to facilitate searches). This will open a page with a table of contents that includes the heading “EduMod for Classroom lab001 and PCs at home.” Clicking that link gets to the series of student EduMod pages that were in use at UCI in fall, 2009. To create “edit windows” whose names will automatically appear in the index for their wiki page requires only the creation to paired headers for programs pasted to R from the first edit window and for results pasted from R back to the second edit window, thus, in general outline (with specific content labels provided by the students and instructor):

```
=| Program edit window 1 (Unrestricted Model Eff and Dow)=
... R program pasted here to R
=| Results edit window 1=
... Results pasted here from R
=| Program edit window 2 (Unrestricted Model for a new Depvar)=
... R program pasted here to R
=| Results edit window 2=
... Results pasted here from R
=| Program edit window 3 (Unrestricted Model) =
... R program pasted here to R
=| Results edit window 3=
```
... Results pasted here from R
=A| Program edit window 4 (Unrestricted Model) =
... R program pasted here to R
=B| Results edit window 4=
... Results pasted here from R

Box 3

The code for the Eff-Dow Restricted Model is usually pasted into one of the first edit windows, as above. Once saved, a clickable edit button will appear to the right of the edit window label. The student then creates another edit window just below where results of running the Restricted Model are posted. Once the edit windows are installed in this way, editing is done only within a single window so that the content of another window is not mistakenly edited. Editing instructions are found at the wiki home page. Editing requires that the student log-in under their real name (enabling linkages through [[pagename link]]) navigation (i.e., the name of the target page in double square brackets).

Don’t set all these windows up before you have a program to start with, and once you do, just add windows as you need them. And don’t plan to subtract one nonsignificant or high VIF variable at a time, Its adding variables to the Unrestricted Model that you should do one at a time, not subtracting later to get to the Restricted Model. (This is a reminder to an undergraduate to read instructions more carefully).

If each header for the A|program/B|results pairs have, in their title, a leading A| or B| to distinguish the two, and the rest of the title is descriptive content, then it is easy to keep track of the development of (1) a single Unrestricted model with a new dependant variable than that of Eff and Dow and (2) successive refinements for the Restricted Model for independent variables that predict the dependant variable. These steps require various types of editing for step (1) and the successive steps in (2).

Once Program 1 runs correctly and you are doing Restricted mode (no new independent variables), you can make changes in and run Program 2 separately. Hence the A| Program ... windows 3 and 4 above need only include the Eff and Dow code for Program, which will run by itself if Program 1 has already been run in the same session with R.

**The importance of debugging the programs within Edit Windows**
The educational use of the program MODification (EduMod) process is complex. Students must keep track of successive modifications of their programs, must save working versions of each of their modifications, and be able to backtrack in the case of errors after making changes in the previous programs. They may also want to backtrack to a much earlier version, and hence need to document in the headers to their EduMod wiki pages how that page or program differs from others.

**The Cardinal Rule of Editing Programs**
The key to EduMod wiki editing is always to save the last working edit of a program rather than try to make further changes in it. That is, never edit a program edit window if the program has run successfully (usually followed by an edit window for the results). Rather,
start a new page and copy the working program to edit in this new edit window so that the older program is preserved.

Order of Editing
The general procedure is:
1. Begin with Eff and Dow’s code or its revised version (at EduMod-1 on the wiki). It computes a Restricted Model with only the (few) significant variables. Test whether the program runs. If not, see Debug.
2. Now, to create the Unrestricted Model for Eff and Dow’s example (depvarname<- “Child Value”, copy the contents of the Unrestricted (xUR<-) into the Restricted (xR<-) model. This is described in Eff and Dow (2009:Figure 3). This is done because all the program output comes from the xR<- section of code (use find: xR<- on the wiki code page), so that to get the Unrestricted (xUR<-) model the contents of that model, beginning and ending for (xUR<-) with the code in box 4. Here lm(depvar~ {list of independent variables}) initiates OLS linear regression.

(Note the correction in red for xUR three lines above, an important distinction)

```
<-lm(depvar~fyll+fydd+dateobs+
cultints+roots+cereals+gath+plow+  
...  
migr+brideprice+nuclearfam+pctFemPolyg
,data=m9)
```

3. Thus, in step 1 (the Eff-Dow original code) the xR<- code has only a few (Restricted) variables, while in step 2 the xUR<- and xR<- codes become identical and the xUR<- codes are Unrestricted, with many variables. A sample of the number of codes in xR<- as it goes through changes is shown in Fig. 1.

(Note the correction in red for xUR two lines above, an important distinction)
Fig. 1: Example of the number of independent variables in $xR<-\text{ at each step editing } xR<-$, Eff and Dow's Restricted Model. (When new independent variables are added to $fx<-$, they are also be added to $\text{indpv}<-$, $xUR<-\text{ and } xR<-\text{)}$

4. In this example, at steps 3-8, variables are removed from $xR<-\text{ at each step of editing and new results are saved, the =A| program edit window= and =|B results edit window = doubling with each step. Many variables may be taken out of $xR<-\text{ at a single step and they do not need to be taken out of other parts of the program. There are two reasons to remove variables from } xR<-\text{ at any given step:}$
   a. The VIF for some variables is over 4. Always do this first before omitting variables that are nonsignificant:
   b. The significance of some variables is larger than $p\text{-value}= .10$.
   c. Do not remove the full or fydd (autocorrelation) variables until needed in your last step (here: step 15) because $p\text{-value} > .10$.

5. At steps 9-11 in this example, all the high VIF and high p-value variables have been removed from the Restricted Model ($xR<-\text{ code, and new independent variables are added. This is a very complex process that requires great care and attention and independent variables should be added only one at a time. Adding more variables at once is very like to create errors that are difficult if not impossible to debug.}$

6. At steps 11-15 there is mostly elimination of variables in the $xR<-\text{ code only, working toward a Restricted Model of predictive variables (keeping full and fydd to the very last step), but also adding new independent variables as new hypotheses suggest:}$
Some may be variables already defined in `fx<-` (the data frame that uses the SCCS variables read from SCCS.Rdata to define independent variables for the study) and simply added by name to `xR<-` (and not elsewhere since they are already defined).

One new variable at a time might be added to `fx<-` (and then must be added by name to `fx<-, indpvx<-, xUR<-, and xR<-`).

**VIF (Variable inflation)**

Anthon Eff (personal communication), notes two considerations for eliminating high VIFs:

1. If two variables are measuring the same thing, then they should be combined, or one removed. For example, if you have two variables that are both serving as measures of the size of extended families, then you could standardize them and take the mean, or you could just drop one of them (a formal test for choosing the best one is the J-test, but you could just try each without the other and see which you prefer).

2. If the variables measure different things, but are highly collinear, you shouldn't drop one, since that would introduce omitted variable bias. If the variable coefficients are insignificant, and the Wald test for dropping them (along with the other variables in the "dropt" list) has a p-value greater than .05, then you don't really have a problem, since they are not in the final model. Likewise, if they are both significant, then you don't have a problem. The problem is when you try to drop them and the Wald test p-value falls below .05, rejecting the null hypothesis that the excluded variables have coefficients equal to zero. In that case, one invokes the high VIFs to explain why these apparently insignificant variables are included in the final model.

**Shortcut to adding new independent variables**

One way to help insure that the choice of a new independent variable will lead to an additional predictor of the dependent variable is to use a Goggle Scholar search: “Standard Cross-Cultural Sample”+yourdeprevarname in ordinary English, e.g., +warfare. This will insure that you retrieve scholarly authors who have investigated your topic using the SCCS database. SCCS+warfare might also work in Goggle Scholar but the hit rate is lower (283 compared to 355, about 25% lower for warfare as a topic).

Another way to help the search for significant predictors, once the R program is running (hence all the variables from SCCSS$, the database), is to perform a cross-tabulation and significance test between two variables, as in this crosstab of v667 (“rape”) and v666 (“interpersonal violence”). Although the significance may be wildly exaggerated, the high significance for the variables may indicate that one will predict the other in 2SLS (in this study “rape” was the dependent variable).

```r
library(gmodels)
setwd("c:/My Documents/MI")
load("SCCS.Rdata",.GlobalEnv)
tab=cbind(SCCS$v667,SCCS$v666)
tabl<-na.omit(tab)  #eliminate cases with missing data
x=tabl[,1]  #take variable for those cases
```

Box 5


\[ y = \text{tabl}, [2] \] # take variable for those cases

CrossTable(x, y, prop.r=FALSE, prop.c=FALSE, prop.t=FALSE, expected=TRUE)

**Debugging**

A depvar may be one that is also defined as an independent variable in `fx<-` without creating a program error. No independent variable may be defined twice under different names, however: This will result in a program error. When adding a new independent variable to `fx<-` (and then elsewhere), take care that this SCCS$ number or category, if specific categories are used (like `cereals=(SCCS$v233==6)*1`, which creates a dichotomous variables for cereals/no cereals while preserving missing data) does not define the same variable twice. It is common, however, to use two distinctively defined categories from the same SCCS$ variable, like that for `cereals` and `roots=(SCCS$v233==5*1`, since these are independently defined.

Our initial testing of the program (box 2) left off in **R code for SCCS** just before the Program 1 segment for:

```
#-----------------------------
#----Multiple imputation----
#-----------------------------
```

1. If the initial pre-imputation lines of code to this point have been successful, then the imputation lines of code can be copied and pasted into R down to the R heading for Program 2 in box 7.
2. After this pre-imputation part of the program finishes, the student can inspect whether errors have occurred by moving the slider in the R window up to see if there are error messages. If there are errors the student should shift to inspecting what the first of the serious errors could mean. If you don’t see how to fix the error, copy the code just above the error and the error itself to the top of your program edit window and contact your instructor is you have trouble fixing it.
3. If there are no errors found in a careful back-tracing of the program’s pre-imputation execution, the Multiple imputation parts of the program (with take 3-5 minutes and will obliterate earlier error messages) can be copied and pasted to the R window, up to the code beginning with

```
#MI--estimate model with network-lagged dependent variables, combine results Box 7
```

4. The multiple imputation part of the code will take considerable time to run, and the student can do something else for five minutes or so. Lines numbered 1-100 and 1-10 will recur as ten imputations are made for each missing value for each of the independent variables specified in the program. The imputation part of the program ends with:

```
> #--impdat is saved as an R-format data file--
```
5. Program 2 can be copied and run up to box 9, and will enable checking whether the depvar is properly defined (check the result against the SCCS codebook entry for the depvar, original identified by number in fx<-

```
summary(depvar)
table(depvar)
```

6. This portion of the code can be checked for errors. Portions of Program 2, such as the contents of, indpv<-, xUR<-, and xR<-, can be individually copied and pasted into R to check for errors. That is, Program 2, however, cannot be run in larger segments, except for the segments that begin and end like these:

```
xUR<-lm(depvar~fyll+fydd+dateobs+
cultints+roots+cereals+gath+plow+
hunt+fish+anim+pigs+milk+bovines+tree+foodtrade+foodscarc+
+ecorich+popdens+pathstress+exogamy+ncmallow+famsize+
sette+localj+h+superj+h+moral+ gods+fempower+femsubs+
+sexratio+war+himilexp+money+wagelabor+
migr+brideprice+nuclearfam+pctFemPolyg
, data=m9)
or
xR<-lm(depvar)~ ffll+ffdd
... , data=m9)
```

7. Program 2 contains a loop that calculates the statistical tests for each of the ten datasets for which missing data have been separately imputed. Thus, attempts to run longer portions of Program 2 will fail (and the errors will be meaningless in this context) without including the full loop that begins and ends with the following statements:

```
#------------------------------------------------------
#---Estimate model on each imputed dataset-----------------
#------------------------------------------------------

... #--loop through the imputed datasets--
for (i in 1:nimp) {...} includes aggregating each separate using...

#------------------------------------------------------
#--Rubin's formulas for combining estimates--
#------------------------------------------------------
```
8. If program 1 and program 2 run without error, the latest version of the program will show the following results. The \( n_n \) values are those of the depvar, the numbers below are the numbers of societies with each value, and the number \( N=171 \) is the sum of the frequencies of each category in the variables. This is the number of societies with nonmissing values (\( N=171 \) of 186 in the SCCS, with 15 societies having missing values). The intercept and regression coefficients are shown for each variable, along with the significance test (p-value) and the VIF (variable inflation factor) for each variable. Here, the p=0.026 for variable \( \text{fyll} \) shows significant clustering of the “child value” dependent variable according to language similarities.

#new Commands:

\[
\text{nn}= \text{Depvar: Values, 'Freqs', N=, and depvarname} \\
\text{aaa}<-c(\text{table(depvar), NROW(depvar), depvarname)} \\
\text{nn} \\
\text{aaa} \\
\text{#Output} \\
\text{>nn} \\
"\text{Depvar: Values | Freqs, and N=}" \\
8  10  12  14  16  17  18  19  20  22  23  24  26  28  30  32  34  36 \\
2   2   1   3   6   1  12  2  25  19  1  28  18  19  7  17  2  6  171 \\
\text{> aaa} \\
8 10 12 14 16 17 18 19 \text{"2" "2" "1" "3" "6" "1" "12" "2"} \\
20 22 23 24 26 28 30 32 \text{"25" "19" "1" "28" "18" "19" "7" "17"} \\
34 36 \text{"2" "6" "171" "child value" Unrestricted Model} \\
\text{> bbb} \\
\text{coef  Fstat    ddf  p-value   VIF} \\
(\text{Intercept}) -5.894 0.113  2212.372  0.737 NA \\
\text{fyll}  2.011 4.954  2814.957  0.026 4.242 \\
\text{fydd} -0.631 1.239 26292.315  0.266 3.557 \\
\text{cultints} 1.057 3.553  8479.664  0.059 5.226 \\
\text{roots}  -4.575 3.732 48226.830  0.053 5.109 \\
\text{cereals} -1.443 0.387 76193.979  0.534 7.452 \\
\text{gath}  -0.449 0.798 76193.979  0.372 3.214 \\
\ldots \]

9. To specify a new dependent variable, Program 2 is changed. Like other program changes this is done by:
   a. Creating a new edit window in which to copy the full set of results (including output labeled in chunks and ending as shown below (the p-values under \text{ccc} are for tests of how well the Restrict Model, when completed, is performing (see Eff and Dow 2009)).

\[
\text{> aaa} \\
\text{Box 12} \\
\]
> bbb
> ccc

<table>
<thead>
<tr>
<th></th>
<th>Fstat</th>
<th>df</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESET</td>
<td>2.668</td>
<td>609.713</td>
<td>0.103</td>
</tr>
<tr>
<td>Wald on restrs.</td>
<td>34.675</td>
<td>167.412</td>
<td>0.000</td>
</tr>
<tr>
<td>NCV</td>
<td>3.174</td>
<td>1383.678</td>
<td>0.075</td>
</tr>
<tr>
<td>SWnormal</td>
<td>0.623</td>
<td>578.843</td>
<td>0.430</td>
</tr>
<tr>
<td>lagll</td>
<td>1.324</td>
<td>106244.100</td>
<td>0.250</td>
</tr>
<tr>
<td>lagdd</td>
<td>3.682</td>
<td>14167.910</td>
<td>0.055</td>
</tr>
</tbody>
</table>

b. Even if these results appear, it is possible that errors occurred after Program 2 was changed, so it is best to check whether the name of the depvar is correct or find an click the OLSresults.csv file (which opens in Excel) in the "C:/My Documents/MI" directory to see if its time stamp corresponds to the time the run finished. If it does, it contains the data that appeared in the R window.
c. In the R output, as in Box 13, you can ignore the messages that were generated when the results were copied into the OLSresults.csv file in the "C:/My Documents/MI" directory.
d. If the aaa-ccc results are not correct, then the problem of where the first error occurred should be addressed using the R slider to view the earlier execution of the program.
e. Each time the R program halts and you begin a new program or program segment, you should leave a whole page of “return/enter” blanks to visually separate the earlier execution lines from the next lines that will be executed. Otherwise it will not be easy to tell where the most recent execution lines begin after the earlier execution segment.

**Naming and defining of (existing or new) independent variables from the SCCS$ dataset**

Typical naming and coding options from Eff and Dow (2009) are shown here:

```r

cereals=(SCCS$\text{v233}==6)*1,gath=SCCS$\text{v203},hunt=SCCS$\text{v204},
fish=SCCS$\text{v205},anim=SCCS$\text{v206},femsubs=SCCS$\text{v890},
pigs=(SCCS$\text{v244}==2)*1,milk=(SCCS$\text{v245}>1)*1,plow=(SCCS$\text{v243}>1)*1,
bovines=(SCCS$\text{v244}==7)*1,tree=(SCCS$\text{v233}==4)*1,
foodtrade=SCCS$\text{v819},foodscarc=SCCS$\text{v1685},
ecorich=SCCS$\text{v857},popdens=SCCS$\text{v156},pathstress=SCCS$\text{v1260},
```

SCCS is the name of the SCCS.Rdata file in which the SCCS variables are stored, and SCCS$ followed by a variable name will retrieve a variable. Most of the SCCS variables are named by number in the SCCS codebook, preceded by “v”, hence “v244” is the variable for types of domesticated animals, and `pigs=(SCCS$\text{v244}==2)*1` is a dichotomized variable for pigs/no pigs, preserving missing data.
**Recoding Variables**

What if you want 4 and 5 versus other categories for a given variables as a dichotomy? E.g. for popdens=SCCSS$v156, the following with contrast the middle range population densities with higher and lower densities:

\[
\begin{align*}
z_4 &\leq (\text{SCCSS}$v156==4) \times 1 \\
z_5 &\leq (\text{SCCSS}$v156==5) \times 1 \\
z_4 &\leq z_4 \\
z_5 &\leq z_5 \\
z_{45} &= z_4 + z_5
\end{align*}
\]

How do you create a dichotomy from Games v239 for societies with games of Strategy versus No strategy? The following will dichotomize so that \( z \) has values 0, 1, and missing data.

\[
\begin{align*}
\text{strat}_4 &\leq (\text{SCCSS}$v239==4) \times 1 \\
\text{strat}_6 &\leq (\text{SCCSS}$v239==6) \times 1 \\
\text{strat}_8 &\leq (\text{SCCSS}$v239==8) \times 1 \\
\text{strategy} &= \text{strat}_4 + \text{strat}_6 + \text{strat}_8 \\
z &\leq (\text{SCCSS}$v777==4) \times 1
\end{align*}
\]

The following will dichotomize the “Child value” depvar from Eff and Dow’s program at 35.

\[
\text{ch} \leq (\text{depvar}>=35) \times 1
\]

It is useful to consult http://www.statmethods.net/management/operators.html, the list of operators in R, for other options, e.g., while \( \geq \) is GE, \( \leq \) is LE and \( \== \) is EQ.

**Comparison of Inferential and (Non-) Descriptive Statistics**

Dow, Burton and White (1982) showed by simulation that linguistic or distance clustering of similarities among survey cases fundamentally alters the variance estimates of regression coefficients and their significance tests. UCI student Amanda McDonald, for example, showed in her powerpoint presentation for the class, with help from the instructor, the following descriptive statistics for the relationship between each of her two independent variables and her dependent variable: Rape, as coded by Sanday (1981). The chi-squared test shows significance values of \( p = .0001 \) and \( p = .0006 \) but these are not really descriptive of the true relationships, since her 2SLS (two-stage OLS regression), which separates out autocorrelation from causal effects, shows values 240 and 11 times less significant: for these variables: \( p = 0.024 \) and \( p = 0.007 \) in the 2SLS.
This order of magnitude of overestimating significance – 10 to 100 times or more is typical of cross-cultural studies using (non-) descriptive statistics. For this reason, the entire literature is riddled with holes, with false indicators of statistical significance.

(Non) Descriptive and Inferential Statistics Compared
Box 19 shows the chi-squared test result of student Nathan Gallagher’s test of Roberts et al. (1959, 1962) expressive culture theory that games of strategy are modeled on and a model of competition and stratification. Here the column variables is SCCS$v270, social stratification, and the row variable is a dichotomy of (1) societies with games of strategy (SCCSS$v239) versus (0) no games of strategy.

```
tab=cbind(strategy,stratif)
tabl<-na.omit(tab)  #eliminate cases with missing data
x=tabl[,1] #take variable for those cases
y=tabl[,2] #take variable for those cases
CrossTable(x,y, prop.r=TRUE, prop.c=FALSE, prop.t=FALSE, expected=TRUE)
```

```
<table>
<thead>
<tr>
<th>y</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>49</td>
<td>24</td>
<td>1</td>
<td>12</td>
<td>5</td>
<td>91</td>
</tr>
<tr>
<td></td>
<td>38.787</td>
<td>21.631</td>
<td>0.746</td>
<td>17.156</td>
<td>12.680</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2.689</td>
<td>0.259</td>
<td>0.087</td>
<td>1.549</td>
<td>4.652</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.538</td>
<td>0.264</td>
<td>0.011</td>
<td>0.132</td>
<td>0.055</td>
<td>.746</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>5</td>
<td>0</td>
<td>11</td>
<td>12</td>
<td>31</td>
</tr>
<tr>
<td></td>
<td>13.213</td>
<td>7.369</td>
<td>0.254</td>
<td>5.844</td>
<td>4.320</td>
<td></td>
</tr>
<tr>
<td></td>
<td>7.894</td>
<td>0.762</td>
<td>0.254</td>
<td>4.548</td>
<td>13.656</td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.097</td>
<td>0.161</td>
<td>0.000</td>
<td>0.355</td>
<td>0.387</td>
<td>.254</td>
</tr>
<tr>
<td>Col.Total</td>
<td>52</td>
<td>29</td>
<td>1</td>
<td>23</td>
<td>17</td>
<td>122</td>
</tr>
</tbody>
</table>
```
Pearson's Chi-squared test

\[ \chi^2 = 36.35028 \quad \text{d.f.} = 4 \quad p = 0.000000245 \]

In Gallagher’s chi-squared test of the Roberts model the significance test is spuriously high (\( p = 0.000000245 \)) compared to 2SLS where \( p < .001 \). (The command "round(object, #digits)" allows one to set the number of decimal places. \( \text{pval} \leftarrow \text{round(pval,} 8 \right) \) would give object \( \text{pval} \) 8 decimal places – thus \( \text{bb} \leftarrow \text{data.frame(round(cbind(mnb,fst,v,pval),3))} \) in the program is set to \( \text{bb} \leftarrow \text{data.frame(round(cbind(mnb,fst,v,pval),8))} \) – Anthon Eff). Then simply rerun the very end of the program (outside the program loop), from the following to the end:

Box 20

#--------------------------------------------
#--Rubin's formulas for combining estimates--
#--------------------------------------------

**Negative Autocorrelation, Hyper-independence and Hyper-significance**

Roberts’ \( \text{pvalue} \) in 2SLS, \( p < 0.000000001 \), has eight significant digits, 100 times more significant than the six digits of the \( \text{pvalue} \) for the chi-squared test for the same correlation, between games of strategy and social stratification. This is a strong reversal of the usual difference. It is not anomalous, however, but reflects the fact that un this 2SLS model, the \( fyll \) language autocorrelation coefficient is negative, \(-0.3573787\). Even though not significant (\( \text{pvalue}=0.165 \)), the negative autocorrelation represents a more diverse dispersal, or hyper-independence of games of strategy. Expressive culture such as games, can be independent invented without material costs, so to speak, because it is a product of imagination, not infrastructure. The explanation for hyper-independence of games of strategy in particular may be that the are associated with more complex or urban societies which do not cluster together but from center-periphery structures.

Other examples, such as student Lyndon Forrester’s study of wealth (\( v1721 \)) as a dependent variable, show that although \( fyll \) and \( fydd \) autocorrelation variables may not have significant \( p \)-values they still have high Stage 1 autocorrelation effects on the estimation, as reflected in their \( R^2 \) values. This is important in terms of the problem of valid estimate of \( p \)-values and \( R^2 \) given Galton’s problem.

There is reason then to retain the \( fyll \) and \( fydd \) language and distance autocorrelation coefficients in 2SLS models because they correct for even the modest effects of autocorrelation on the significance estimates for other variables.
The Standard Cross-Cultural Sample and GIS

As noted, the SCCS database contains survey and geographically resolved data from the Standard Cross-Cultural Sample collaborative project accessible to researchers, students, and classrooms. A GIS server is also available for creating maps of the distributions of SCCS societies colored by their features on individual SCCS variables.

Anthropology and the end of a positivist fantasy

That E. B. Tylor (1889) was roundly criticized by Sir Francis Galton for his assumptions about descriptive statistics – the mistaken assumption that correlation equals causation, for example – has never been forgotten in arguments over anthropological theory. Galton noted that since Tylor’s cases were not independent but multiple nonindependent samples from regions of the world where societies influenced one another through borrowing, trading, attacking, peacemaking, and splitting off from one another from common ethnic, linguistic, political and religious groups.

Anthropologists today have split into different groups over this issue. In cultural anthropology the predominant view is that it is ok to use descriptive statistics for household censuses and inventories of the features of individual communities but not to make inferences about larger historical or functional relations from correlations about samples of different societies. This majority view is critical of what is called the “HRAF” method of comparison, in which inferences are made about functional relations among features of different societies from correlations based on samples that are drawn by methods of probability sampling from a larger universe. It is well known that by such means (Kish 1965) it is possible to get an unbiased estimate of the mean of the larger universe of cases from which the sample is drawn, and that the error bounds of these descriptive means can be accurately estimated. The descriptive mean and standard deviation is derived from the assumption that cases are sampled independently, i.e., without a sampling bias.

This is quite different from Galton’s problem, however, in which even with a complete population, and no sampling whatsoever, the cases are not historically independent. This is the problem of autocorrelation, which may take the form of historically replicated copies of the same originals, borrowing or diffusion, similarities or differences produced through interaction or splitting off from the same historical antecedents. The “HRAF” brand of contemporary positivism (valuing or prescribing preferred methodology above logically scientific practice) prefers not to recognize the problem as posed by Galton. As distinguished a methodological source as Bernard’s Handbook (1998:678-679) allows contradictory statements by its authors. First (p. 678): “We suggest that those who worry about Galton's Problem misunderstand two requirements of statistical inference—that sample cases be independent and that the measures on them be independent…. Independence of cases means only that the choice of one case is not influenced by the choice of any other case (which random sampling guarantees).” Second (p. 678-79): “whether or not you worried about Galton's Problem made a big difference in how you would do a study. Naroll’s tests for the possibility of diffusion were quite time consuming to carry out. This was probably why most
cross-culturalists altered their sampling strategy so as to eliminate multiple cases from the sample culture area.” These authors are conflating the two meanings on nonindependence.

Bernard’s *Handbook* authors, however, seem to reconcile these two contradictory views as follows (p.679): “Recently, however, mathematical anthropologists have developed statistical solutions and computer programs that treat the proximity of societies (in distance or language) as a variable whose influence can be treated in a multiple regression analysis. (This is called testing for spatial autocorrelation.\(^4\) (footnote 1: For the newer treatments of Galton’s problem, see Burton and White (1991) for references; see also Dow (1991).

Correlations between variables do not represent functional relations within societies but need to be partitioned into parts: from common historical origin, commonality through the splitting of groups with a common language history into distinct societies, spatial diffusion, and a residual of possible functional relationships. DRW: I don’t know how this is done, but perhaps using an lm(depvar<-…) with no fyll or fydd provides a comparison. (Anthon?)

**The EduMod Course Description**
Run a powerful regression program that does estimates of the effects of causal factors that you select for study in a database of two thousand variables on features of the full range of diverse human cultures (the SCCS). Take the open source page R home with you and the database and software. Receive tutorials to learn these methods, learn from other student projects, do an 8-minute powerpoint on your findings, and finish with a 12 page term paper done with writing-course critiques. Learn how to understand complex human systems and write about them.

**References**


\(^4\) For the newer treatments of Galton’s problem, see Burton and White (1991) for references; see also Dow (1991).


