Novometric Analysis of Transition Matrices to Ascertain Markovian Order

Paul R. Yarnold, Ph.D.
Optimal Data Analysis, LLC

The American National Election Panel Study modeled transitions in social class identification occurring between 1956, 1958, and 1960. Visual analysis suggested “…that transition probabilities linking class identifications in 1958 and 1960 vary with 1956 identification” (italics added). Transition tables were compared using Goodman’s chi-square procedure: $\chi^2=98.2$, df=2, $p<0.0001$. Based on visual examination and non-disentangled omnibus chi-square findings, the null hypothesis of a first-order Markov model was rejected in favor of the alternative hypothesis that the underlying temporal process is a second-order Markovian (pp. 13-15). In contrast, novometric analysis indicated that a first-order Markov model is appropriate for these data.

Data used in this application are presented in Table 1 (SAS™ code used to construct the data set is given in the Appendix). Social class identification in each year was dummy-coded as Working=1, and Middle=2, for all observations.

Table 1: Social Class Identification Transitions Across 1956, 1958, and 1960 Measurements

<table>
<thead>
<tr>
<th>1956</th>
<th>1958</th>
<th>Middle</th>
<th>Working</th>
</tr>
</thead>
<tbody>
<tr>
<td>Middle</td>
<td>Middle</td>
<td>216</td>
<td>70</td>
</tr>
<tr>
<td>Working</td>
<td>Middle</td>
<td>56</td>
<td>75</td>
</tr>
<tr>
<td>Middle</td>
<td>Working</td>
<td>42</td>
<td>92</td>
</tr>
<tr>
<td>Working</td>
<td>Working</td>
<td>47</td>
<td>549</td>
</tr>
</tbody>
</table>

The first analysis treated social class in 1956 and 1958 as attributes (consistent findings resulted if attributes were treated as ordered or categorical), and social class in 1960 as the class variable. An exploratory enumerated-optimal classification tree analysis (EO-CTA) identified a single optimal model: if 1958 social class = Middle, predict 1960 social class = Middle; otherwise predict 1960 social class = Working. The confusion matrix for this model in training (and in jackknife) analysis is presented in Table 2: as seen, the model correctly predicted the actual class status of 7 of 8 observations self-identified as Working class, and of 6 of 8 observations self-identified as Middle class.

Table 2: Confusion Table for EO-CTA Model

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
<th>Sensitivity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Working</td>
<td>641</td>
<td>81.6</td>
</tr>
<tr>
<td>Middle</td>
<td>89</td>
<td>75.4</td>
</tr>
</tbody>
</table>
Because this was the only optimal model identified, it thus is also the globally-optimal (GO) model in this application.\textsuperscript{7,11} The model was statistically reliable (exact $p<0.0001$); it yielded relatively strong ESS=56.9 (for 10,000 bootstrap iterations, exact discrete 95% CI for model ESS=50.6-63.2; for 10,000 Monte Carlo experiments, exact discrete 95% CI for chance ESS=0.1-5.8); and had stable classification accuracy in training and leave-one-out (one-sample jackknife\textsuperscript{19-21}) analysis. For this model D=1.51 (exact discrete 95% CI=1.17-1.95).\textsuperscript{7,11}

In contrast to the conclusion reached on the basis of visual examination and omnibus chi-square—that 1960 class status identification was related to both 1956 and 1958 self-classifications (suggesting a second-order Markovian), novometric analysis found that only the self-classifications recorded in 1958 predicted the self-classifications that were made two years later. The novometric result thus supports the use of a first-order Markov model.

Accordingly, a second novometric analysis was conducted predicting 1958 class self-identification (class variable) as a function of 1956 self-identification (attribute). As before, treating the attribute as ordered or categorical did not affect the result. And, conducting a confirmatory analysis—replicating the prior model, versus an exploratory analysis, also did not affect results presently. Using EO-CTA a single GO model was identified: if 1956 social class=Middle, then predict 1958 social class=Middle; otherwise predict 1958 social class=Working.

Table 3 is the confusion matrix for this model in training and jackknife analysis: the model correctly predicted the actual class status of 7 of 8 observations self-identified as Working class, and 7 of 10 observations as Middle class. The model was statistically reliable (exact $p<0.0001$); achieved moderate-to-relatively strong ESS=50.2 (for 10,000 bootstrap iterations, exact discrete 95% CI for model ESS=43.9-56.5; for 10,000 Monte Carlo experiments, exact discrete 95% CI for chance ESS=0.1-5.8); and had stable classification accuracy in training and leave-one-out analysis. For this model D=1.98 (exact discrete 95% CI=1.54-2.56).

Table 3: Confusion Table for EO-CTA Model

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Class</td>
<td>Working</td>
</tr>
<tr>
<td>Working</td>
<td>596</td>
</tr>
<tr>
<td>Middle</td>
<td>131</td>
</tr>
</tbody>
</table>

The exact 95% CIs for ESS and D of the two novometric models overlap, indicating the models do not differ significantly with respect to predictive accuracy normed against chance (ESS) and also against parsimony (D). Taken together the novometric analyses offer moderate to relatively strong evidence of a reproducible first-order Markov model in this application.

References


Yarnold PR, Soltysik RC (2014). Globally optimal statistical classification models, II: Unrestricted class variable, two or more attributes. *Optimal Data Analysis, 3*, 78-84.


**Author’s Notes**

Analyzed data are publically available, and no conflict of interest was reported.
Appendix

SAS™ Code used to Construct (Reproduce^1) the Data File for Analysis by ODA Software^7,16,17,22,23

data real;
infile datalines;
input 1956 1958 1960;
cards;
1 1 1
;
Data example;
Do n=1 to 216;
  put '2 2 2';
end;

Do n=1 to 70;
  put '2 2 1';
end;
Do n=1 to 56;
  put '1 2 2';
end;
Do n=1 to 75;
  put '1 2 1';
end;
Do n=1 to 42;
  put '2 1 2';
end;
Do n=1 to 92;
  put '2 1 1';
end;
Do n=1 to 47;
  put '1 1 2';
end;
Do n=1 to 549;
  put '1 1 1';
end;
Output;
Run;