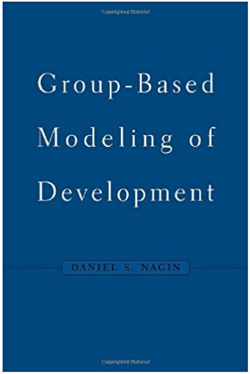
Trajectory analysis

Jan Helmdag
University of Greifswald

September 26, 2017

Nagin, D. (2005). *Group-based modeling of development.* Harvard University Press.



Preliminaries

Prominent procedures for **descriptive** and **exploratory** analysis

- Description of parameters
- Graphical inspection of time series graphs
- Multi-dimensional scaling
- Principal component analysis
- Cluster analysis

Problem: None of these procedures can truly capture the charateristics of time-series data *and* is built on inferential processes

Why?

- Finding subgroups/identifying common trajectories in time-series data
- Growth curve for comparison indivdual vs sample

> *"It makes no sense to assume that everyone is increasing (or decreasing) in depression ... many persons will never be high in depression, others will always be high, while others become increasingly [or decreasingly] depressed" Raudenbush (2001, p. 59)*

- Applications: Modelling 'path dependency', identifying turning points (recession, child-birth, EU withdrawal)
- Guiding question: How to best model population heterogeneity of individual-level trajectories?

How?

- Finite mixture modeling based on maximum likelihood estimation
- Estimating probabilities for group membership
- Choice of parametrical form (normal, logit, poisson, zip)
- Choice of link (polynomials)
  1. x^1 linear (constant, increase, decrease)
  2. x^2 square ([inverse]-u-shape, single peak)
  3. x^3 cubic (two turning points, steep increase or decrease)
  4. x^4 (quartic) and x^5 (quintic) are also possible, but seldomly used

What to do?

Six steps for conducting a trajectory analysis

1. Reshape dataset
2. Conduct analysis and find highest BIC
3. Reduce the number of parameters and re-estimate model (lowering the degrees of polynomials)
4. Create graphs and inspect trajectories
5. Conduct a multinomial logit analysis with predicted groups as dependent variable
6. Calculate marginal effects and predicted probabilities

1^{st} step: Reshape dataset

- Wide format is needed for trajectory analysis
- To reshape an *NxT* dataset into wide format type:
  `reshape wide varlist, i(iso) j(year)`

2^{nd} step: Find highest BIC

- Every model comes with model fit AIC & BIC
- Choose model with highest BIC (Kass and Raftery 1995; Raftery 1995; Schwarz 1978)
- BIC rewards parsimony
- $BIC = log(L) - 0.5 * k * log(N)$
- Advanced: determine Bayes factor with $e^{BIC_i - BIC_j}$ (see Nagin 2005, pp. 68–70)

2^{nd} step: Find highest BIC

- Example 1: Loop for finding lowest BIC:

```
local best_bic -10^99
if ‘e(BIC_N_data)’ > ‘best_bic’ {
local best_bic ‘e(BIC_N_data)’
}
```

- Example 2: Calculating probability for correct model

```
gen nominator = exp(bicb-bicb[_N])
quietly sum nominator
gen denominator = ‘r(sum)’
gen prob_correct = nominator/denominator
```

3rd step: Determine # of polynomials

- Determine shape of polynomials
- t-tests and graphical inspection
- The lower the number of polynomials the better
- In the majority of analyses ‘consistently low’ and ‘consistently high’ groups can be modelled with a constant (and no slope/polynomial)

4^{th} step: Graphs

- traj.ado comes with command for creating graphs:

  ```
  trajplot, ci ytitle("yvar") xtitle("time")
  ```

- Currently work in progress: ado for creating trajectory graphs (Helmdag 2017)

5^{th} step: Multinomial logit

Why mlogit?

- Easily performed
- Standard errors are properly computed
- Correct estimation of parameter variance and covariance (see Nagin 2005, pp. 115–6)

Example with clustered standard errors on unit-level:

- `mlogit depvar indepvars, vce(cluster iso) rrr`

6^{th} step: mlogit post-estimation

Loop over margins and combine them into a single plot

```
forvalues i = 1/4 {
      margins, dydx(indepvar) atmeans predict(outcome(`i')) ///
            saving(marg`i', replace)
}
*ssc install combomarginsplot
combomarginsplot marg1 marg2 marg3 marg4 ///
      , recast(scatter) yline(0)
```

Application

- Simple example with randomly generated dataset ($n = 200, t = 20$)
- Continuous dependent variable with four groups
  1. consistently low at 0
  2. linearly increasing from 1 to 20
  3. linearly decreasing from 20 to 1
  4. consistently high at 20
- Every group has a standard deviation of 2

Application

```
traj, var(y1_*) indep(time*) model(cnorm) ///
      min(-10) max(30) order(3 3 3 3) detail
```

Maximum Likelihood Estimates
Model: Censored Normal (CNORM)

Group	Parameter	Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
1	Intercept	0.01008	0.30999	0.033	0.9741
	Linear	-0.00790	0.12473	-0.063	0.9495
	Quadratic	0.00221	0.01363	0.162	0.8714
	Cubic	-0.00013	0.00043	-0.293	0.7698
2	Intercept	-0.19149	0.30999	-0.618	0.5368
	Linear	1.05909	0.12473	8.491	0.0000
	Quadratic	-0.00573	0.01363	-0.420	0.6743
	Cubic	0.00015	0.00043	0.362	0.7173
3	Intercept	19.91765	0.30999	64.254	0.0000
	Linear	-1.03359	0.12473	-8.287	0.0000
	Quadratic	0.00447	0.01363	0.328	0.7427
	Cubic	-0.00011	0.00043	-0.255	0.7991
4	Intercept	20.01140	0.30999	64.556	0.0000
	Linear	-0.02211	0.12473	-0.177	0.8593
	Quadratic	0.00254	0.01363	0.186	0.8521
	Cubic	-0.00005	0.00043	-0.114	0.9094
	Sigma	2.00188	0.02244	89.230	0.0000
Group membership					
1	(%)	24.99987	3.06914	8.146	0.0000
2	(%)	25.00001	3.06913	8.146	0.0000
3	(%)	25.00008	3.06918	8.146	0.0000
4	(%)	25.00003	3.06913	8.146	0.0000

BIC= -8812.30 (N=4000) BIC= -8782.34 (N=200) AIC= -8749.36 L= -8729.36

Application

Reducing the number of parameters

```
traj, var(y1_*) indep(time*) model(cnorm) ///
      min(-10) max(30) order(0 1 1 0) detail
```

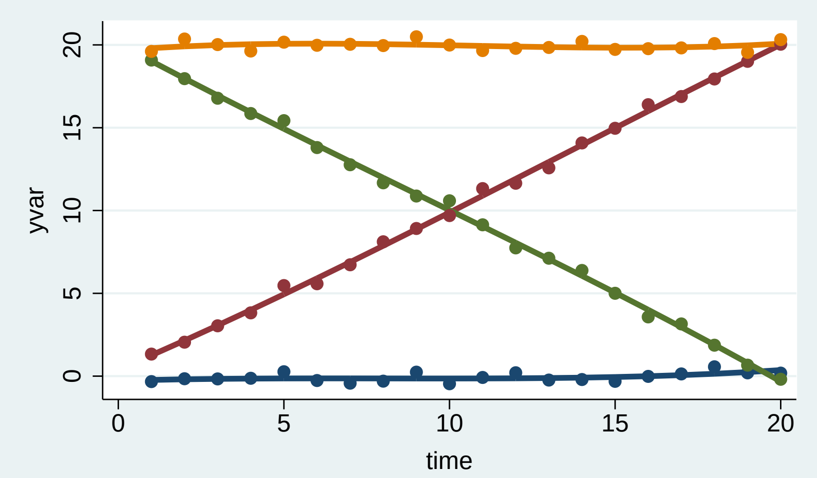
Maximum Likelihood Estimates
Model: Censored Normal (CNORM)

Group	Parameter	Estimate	Standard Error	T for H0: Parameter=0	Prob > |T|
1	Intercept	-0.03201	0.06341	-0.505	0.6137
2	Intercept	-0.04348	0.13172	-0.330	0.7413
	Linear	0.99920	0.01100	90.870	0.0000
3	Intercept	19.77917	0.13172	150.159	0.0000
	Linear	-0.98210	0.01100	-89.315	0.0000
4	Intercept	20.03656	0.06341	316.006	0.0000
	Sigma	2.00281	0.02242	89.342	0.0000
Group membership					
1	(%)	24.99938	3.06524	8.156	0.0000
2	(%)	25.00259	3.06542	8.156	0.0000
3	(%)	24.99922	3.06529	8.156	0.0000
4	(%)	24.99881	3.06528	8.155	0.0000

BIC= -8772.69 (N=4000) BIC= -8757.71 (N=200) AIC= -8741.22 L= -8731.22

Application

```
trajplot, ytitle("yvar") xtitle("time")
```



Application