Online Appendix to accompany:
Sterba, S.K. (2016) Interpreting and testing interactions in conditional mixture models. ***Applied Developmental Science*, 20, 29–43.**

Table of Contents:

Part 1.

Syntax for Example 1: Product-term mixture (LCGA) syntax including probing simple slopes for two 2-way interactions.

```
DATA: FILE = example1.dat;
DEFINE: ga=arrest*gender; sa=arrest*supportc; !create product terms
VARIABLE: NAMES ARE y1-y3 gender absub supportc arrest; !list variables in order from dataset
USEVARIABLES ARE y1-y3 gender absub supportc arrest sa ga; !list variables used in analysis
CATEGORICAL ARE y1-y3; !list categorical outcomes
MISSING ARE . ; !specify missing data code
CLASSES=class(3); !specify name for latent categorical variable (here, class)
                  !specify desired number of classes (here, 3)
ANALYSIS: type = mixture; !specify that model is a mixture
estimator=ml; starts=200 20; algorithm=integration;

 MODEL:
 %overall%
!Regress latent class on each predictor and product term; see equations (8) and (10)
!Parameters that are used in constructing simple slopes are given a label in parentheses, e.g., (BG1).
!Labels in () are arbitrary and are chosen by user.
!Parameters in the last class (here class 3) are fixed =0 for identification & are not mentioned in code
 class#1 on absub;
 class#1 on gender (BG1);
 class#1 on supportc (BS1);
 class#1 on arrest (BA1);
 class#1 on sa (BSA1);
 class#1 on ga (BGA1);
 class#2 on absub;
 class#2 on gender (BG2);
 class#2 on supportc (BS2);
 class#2 on arrest (BA2);
 class#2 on sa (BSA2);
 class#2 on ga (BGA2);
  int by y1-y3@1; !specify intercept
  lin by y1@0 y2@1 y3@2; !specify linear slope
  [y1$1-y3$1@0]; !item thresholds fixed to 0 over time
  int@0; lin@0; int with lin@0; !no random effects for growth coefficients

  %class#1% [int lin]; !allow growth factor means to differ across class; (equation (7))
  %class#2% [int lin];
  %class#3% [int lin];

 MODEL CONSTRAINT: NEW (
  ss1s_a1 ss1s_a0 ss1g_a1 ss1g_a0
  ss2s_a1 ss2s_a0 ss2g_a1 ss2g_a0);!declare names of simple slopes to be formed
```

```
  ss1s_a1=(BS1+BSA1*1); !class 1 (vs. 3): simple slope of support where arrest=1
  ss1s_a0=(BS1+BSA1*0); ! class 1 (vs. 3): simple slope of support where arrest=0
  ss1g_a1=(BG1+BGA1*1); !class 1 (vs. 3): simple slope of gender where arrest=1
  ss1g_a0=(BG1+BGA1*0); !class 1 (vs. 3): simple slope of gender where arrest=0

  ss2s_a1=(BS2+BSA2*1); !class 2 (vs. 3): simple slope of support where arrest=1
  ss2s_a0=(BS2+BSA2*0); ! class 2 (vs. 3): simple slope of support where arrest=0
  ss2g_a1=(BG2+BGA2*1); !class 2 (vs. 3): simple slope of gender where arrest=1
  ss2g_a0=(BG2+BGA2*0); class 2 (vs. 3): simple slope of gender where arrest=0
 PLOT: TYPE IS PLOT1 PLOT2 PLOT3; SERIES IS y1-y3(*); !request plots
```

Note: If one does not want to probe simple slopes, the MODEL CONSTRAINT statement can be deleted. Wald tests described in the paper were implemented using MODEL TEST statements. For instance, to obtain the test result in Table 2 row 2, add the following MODEL TEST statement to the above code, right before the PLOT statement: MODEL TEST: BS1=0; BS2=0;
Finally, note that gender=1 for males and gender=0 for females in Example 1.

Syntax for Example 1: Multiple-group mixture (LCGA) syntax.

Note that the below MG code produces results that pertain to arrest=0 for syntax under the %overall% statement. To obtain %overall% results for arrest=1 change the blue **2** *to* **1**.

```
DATA: FILE = example1.dat;
VARIABLE: NAMES ARE y1-y3 gender absub supportc arrest;
 USEVARIABLES ARE y1-y3 gender absub supportc arrest;
 CATEGORICAL ARE y1-y3; MISSING ARE . ;
 CLASSES = cg (2) class (3);
!specify artificial name for grouping variable (here, cg) with # of observed groups (here, 2)
!specify name for latent categorical variable (here, class) with desired # of classes (here, 3)
 KNOWNCLASS = cg (arrest=0 arrest= 1);
!relate artificial name for grouping variable (here, cg) to name of grouping variable in dataset (arrest)
!list lowest value (here, 0) first followed by higher value (here, 1)
 ANALYSIS: type = mixture;
 estimator=ml; starts=200 20; algorithm=integration;

MODEL:
 %overall%
 class on cg  absub gender supportc; !see equations (9) & (11)
 [cg#1@0.082]; !could estimate--here fixed at observed proportion of arrest; see equation (11)
 int by y1-y3@1; !specify intercept
 lin by y1@0 y2@1 y3@2; !specify linear slope
 [y1$1-y3$1@0]; !item thresholds fixed to 0 over time
  int@0; lin@0; int with lin@0; !no random effects for growth coefficients

 MODEL class:
 %class#1% [int lin]; !allow growth factor means to differ across class (equation (7))
 %class#2% [int lin];
 %class#3% [int lin];

 MODEL cg:
  !specifying %cg#2% here asks for %overall% results, above, to pertain to arrest=0
  %cg#2%
  class on gender supportc;!specify effects on class that differ across observed group (eqn. (9))
```

Note. One may want to use manual start values, in addition to random start values, to ensure that this model and the Example 1 PT mixture both have the same class ordering.

Syntax for Example 2: Product term mixture (LCA) syntax including probing simple slopes for a 3-way interaction.

```
DATA: file is example2.dat;
DEFINE: !create product terms
agemale=age*male;
blackmale=black*male;
ageblack=age*black;
amb=age*male*black;
VARIABLE: NAMES ARE y1-y9 age male black;
USEVARIABLES ARE y1-y9 age male black
agemale blackmale ageblack amb;
CATEGORICAL ARE y1-y9; !list categorical outcomes
CLASSES=class(4); !specify name for latent categorical variable (here, class)
                  !specify desired number of classes (here, 4)
ANALYSIS: type=mixture; !specify that model is a mixture
starts=200 20; estimator=ml;

 MODEL:
  %overall%
!Regress latent class on each predictor and product term; see equations (13) and (10)
!Parameters that are used in constructing simple slopes are given a label in parentheses, e.g., (B_bm1).
!Labels in () are arbitrary and are chosen by user.
!Parameters in the last class (here class 4) are fixed =0 for identification & are not mentioned in code
  class on  male black;
  class#1 on blackmale (B_bm1);
  class#1 on agemale (B_am1);
  class#1 on age (B_a1);
  class#1 on ageblack (B_ab1);
  class#1 on amb (B_abm1);
  class#2 on blackmale (B_bm2);
  class#2 on agemale (B_am2);
  class#2 on age (B_a2);
  class#2 on ageblack (B_ab2);
  class#2 on amb (B_abm2);
  class#3 on blackmale (B_bm3);
  class#3 on agemale (B_am3);
  class#3 on age (B_a3);
  class#3 on ageblack (B_ab3);
  class#3 on amb (B_abm3);

  %class#1% [y1$1-y9$1]; !allow endorsement thresholds for the 9 items to differ across class (eqn. (12))
  %class#2% [y1$1-y9$1];
  %class#3% [y1$1-y9$1];
  %class#4% [y1$1-y9$1];

   MODEL CONSTRAINT: NEW (
   s1a_b1m1 s1a_b1m0 s2a_b1m1 s2a_b1m0
   s3a_b1m1 s3a_b1m0 s1a_b0m1 s1a_b0m0
   s2a_b0m1 s2a_b0m0 s3a_b0m1 s3a_b0m0); !declare names of simple slopes to be formed and tested

  !class 1 vs. 4: simple slope of age at chosen values of race and gender
  s1a_b1m1=(b_a1 + b_ab1*1 + b_am1*1 + b_abm1*1*1);!simple slope for black males
  s1a_b0m1=(b_a1 + b_ab1*0 + b_am1*1 + b_abm1*1*0);!simple slope for white males
  s1a_b1m0=(b_a1 + b_ab1*1 + b_am1*0 + b_abm1*0*1);!simple slope for black females
```

```
s1a_b0m0=(b_a1 + b_ab1*0 + b_am1*0 + b_abm1*0*0);!simple slope for white females

!class 2 vs. 4: simple slope of age at chosen values of race and gender
s2a_b1m1=(b_a2 + b_ab2*1 + b_am2*1 + b_abm2*1*1); !simple slope for black males
s2a_b0m1=(b_a2 + b_ab2*0 + b_am2*1 + b_abm2*1*0); !simple slope for white males
s2a_b1m0=(b_a2 + b_ab2*1 + b_am2*0 + b_abm2*0*1); !simple slope for black females
s2a_b0m0=(b_a2 + b_ab2*0 + b_am2*0 + b_abm2*0*0); !simple slope for white females

!class 3 vs. 4: simple slope of age at chosen values of race and gender
s3a_b1m1=(b_a3 + b_ab3*1 + b_am3*1 + b_abm3*1*1); !simple slope for black males
s3a_b0m1=(b_a3 + b_ab3*0 + b_am3*1 + b_abm3*1*0); !simple slope for white males
s3a_b1m0=(b_a3 + b_ab3*1 + b_am3*0 + b_abm3*0*1); !simple slope for black females
s3a_b0m0=(b_a3 + b_ab3*0 + b_am3*0 + b_abm3*0*0); !simple slope for white females

PLOT: TYPE=PLOT1 PLOT2 PLOT3; SERIES IS y1-y9(*);
```

Note: If one does not want to probe simple slopes the MODEL CONSTRAINT statement can be deleted.

Syntax for Example 2: Multiple-group mixture (LCA) syntax including probing simple slopes for a 3-way interaction.

Note that the below MG code produces results that pertain to male=0 for syntax under the %overall% statement. To obtain %overall% results for male=1 change the blue **2** *to* **1** *and (optionally, also change the red label for the simple slopes from* 0 *to* 1*).*

```
 DATA: file is example2.dat;
 DEFINE: ageblack=age*black;
VARIABLE: NAMES ARE y1-y9 male black age;
 USEVARIABLES ARE y1-y9 age male black ageblack;
 CATEGORICAL ARE y1-y9;
 CLASSES = cg (2) class(4);
!specify artificial name for grouping variable (here, cg) with # of observed groups (here, 2)
!specify name for latent categorical variable (here, class) with desired # of classes (here, 4)
 KNOWNCLASS= cg (male=0 male=1);
!relate artificial name for grouping variable (here, cg) to name of grouping variable in dataset (male)
!list lowest value (here, 0) first followed by higher value (here, 1)
 ANALYSIS: type=mixture; starts=200 20; estimator=ml;

 MODEL:
  %overall%
  class on cg black; !see equations (14) & (11)
  class#1 on age (b_a1);
  class#1 on ageblack (b_ab1);
  class#2 on age (b_a2);
  class#2 on ageblack (b_ab2);
  class#3 on age (b_a3);
  class#3 on ageblack (b_ab3);
  [cg#1@0.048]; !could estimate--here fixed at observed proportion of boys; see equation (11)

  MODEL class:
  %class#1% [y1$1-y9$1]; !allow endorsement thresholds for the 9 items to differ across class (eqn. (12))
  %class#2% [y1$1-y9$1];
  %class#3% [y1$1-y9$1];
  %class#4% [y1$1-y9$1];
```

```
MODEL cg:
!specifying %cg#2% asks for %overall% results, above, to pertain to male=0
 %cg#2% class on age black ageblack; !specify effects on class that differ across obs. group (eqn. (14))

 MODEL CONSTRAINT:
 NEW (s1a_b0m0 s1a_b1m0
           s2a_b0m0 s2a_b1m0
           s3a_b0m0 s3a_b1m0); !declare names of simple slopes to be formed and tested
  s1a_b1m0=(b_a1+b_ab1*1);  !class 1 vs. 4: simple slope of age for black females
  s1a_b0m0=(b_a1+b_ab1*0);  !class 1 vs. 4: simple slope of age for white females
  s2a_b1m0=(b_a2+b_ab2*1);  !class 2 vs. 4: simple slope of age for black females
  s2a_b0m0=(b_a2+b_ab2*0);  !class 2 vs. 4: simple slope of age for white females
  s3a_b1m0=(b_a3+b_ab3*1);  !class 3 vs. 4: simple slope of age for black females
  s3a_b0m0=(b_a3+b_ab3*0);  !class 3 vs. 4: simple slope of age for white females
```

Note: One may want to use manual start values, in addition to random start values, to ensure that this model and the Example 2 PT mixture both have the same class ordering. Also, if one does not want to probe simple slopes the MODEL CONSTRAINT statement can be deleted.

Part. II.

Explanation for Equation (2):

$$\frac{p(c_i = k \mid x_{1i} = x_{2i} = 0)}{p(c_i = K \mid x_{1i} = x_{2i} = 0)} = \frac{\dfrac{\exp(\beta_0^{(k)} + \beta_1^{(k)} 0 + \beta_2^{(k)} 0)}{\sum_{k=1}^{K} \exp(\beta_0^{(k)} + \beta_1^{(k)} 0 + \beta_2^{(k)} 0)}}{\dfrac{\exp(\beta_0^{(K)} + \beta_1^{(K)} 0 + \beta_2^{(K)} 0)}{\sum_{k=1}^{K} \exp(\beta_0^{(k)} + \beta_1^{(k)} 0 + \beta_2^{(k)} 0)}} = \frac{\exp(\beta_0^{(k)})}{\exp(0)} = \exp(\beta_0^{(k)})$$

Where the coefficients in the reference (Kth) class are fixed to 0 for identification.

Explanation for Equation (3):

$$\frac{p(c_i = k \mid x_{1i} + 1, x_{2i})}{p(c_i = K \mid x_{1i} + 1, x_{2i})} \Bigg/ \frac{p(c_i = k \mid x_{1i}, x_{2i})}{p(c_i = K \mid x_{1i}, x_{2i})} =$$

$$\frac{\dfrac{\exp(\beta_0^{(k)} + \beta_1^{(k)}(x_{1i}+1) + \beta_2^{(k)} x_{2i})}{\sum_{k=1}^{K} \exp(\beta_0^{(k)} + \beta_1^{(k)}(x_{1i}+1) + \beta_2^{(k)} x_{2i})}}{\dfrac{\exp(\beta_0^{(K)} + \beta_1^{(K)}(x_{1i}+1) + \beta_2^{(K)} x_{2i})}{\sum_{k=1}^{K} \exp(\beta_0^{(k)} + \beta_1^{(k)}(x_{1i}+1) + \beta_2^{(k)} x_{2i})}} \Bigg/ \frac{\dfrac{\exp(\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2i})}{\sum_{k=1}^{K} \exp(\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2i})}}{\dfrac{\exp(\beta_0^{(K)} + \beta_1^{(K)} x_{1i} + \beta_2^{(K)} x_{2i})}{\sum_{k=1}^{K} \exp(\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2i})}} =$$

$$\frac{\exp(\beta_0^{(k)} + \beta_1^{(k)}(x_{1i}+1) + \beta_2^{(k)} x_{2i})}{\exp(\beta_0^{(k)} + \beta_1^{(k)} x_{1i} + \beta_2^{(k)} x_{2i})} = \frac{\cancel{\exp(\beta_0^{(k)}) \exp(\beta_1^{(k)} x_{1i})} \exp(\beta_1^{(k)}) \cancel{\exp(\beta_2^{(k)} x_{2i})}}{\cancel{\exp(\beta_0^{(k)}) \exp(\beta_1^{(k)} x_{1i}) \exp(\beta_2^{(k)} x_{2i})}} = \exp(\beta_1^{(k)})$$

Where the coefficients in the reference (Kth) class are fixed to 0 for identification.