Mixture Models with Covariates & Distal Outcomes

Adam Garber

Norwegian University of Science and Technology - A Course in `MplusAutomation`

June 01, 2021

Preparation

Data source:

1. The first example utilizes a dataset on undergraduate *Cheating* available from the `poLCA` package (Dayton, 1998): See documentation here
2. The second examples utilizes the public-use dataset, *The Longitudinal Survey of American Youth* (**LSAY**): See documentation here

---

Load packages

```
library(naniar)
library(tidyverse)
library(haven)
library(glue)
library(MplusAutomation)
library(here)
library(janitor)
library(gt)
library(poLCA)
```

Incorporating distal outcome variables with mixture models

Note: Prior to adding covariates or distals enumeration must be conducted.

See Lab 7 for examples of enumeration with MplusAutomation

`DU3step` auxiliary variable integration

- Using the `DU3step` you can specify distal relations but cannot specify models with covariates & distals

---

Application: Undergraduate Cheating behavior

"Dichotomous self-report responses by 319 undergraduates to four questions about cheating behavior" (poLCA, 2016).

Prepare data

```
data(cheating)

cheating <- cheating %>% clean_names()

df_cheat <-  cheating %>%
  dplyr::select(1:4) %>%
  mutate_all(funs(.-1)) %>%
  mutate(gpa = cheating$gpa)
```

Run the **DU3step** model with `gpa` as distal outcome

```
m_stepdu  <- mplusObject(
  TITLE = "DU3STEP add distal GPA",
  VARIABLE =
   "categorical = lieexam-copyexam;
    usevar = lieexam-copyexam;
    auxiliary = gpa (du3step);
    classes = c(2);",

  ANALYSIS =
   "estimator = mlr;
    type = mixture;
    starts = 500 100;
    processors = 10;",

  OUTPUT = "sampstat patterns tech11 tech14;",

  PLOT =
    "type = plot3;
     series = lieexam-copyexam(*);",

  usevariables = colnames(df_cheat),
  rdata = df_cheat)

m_stepdu_fit <- mplusModeler(m_stepdu,
                dataout=here("20-three-step", "du3step_mplus", "lca_du3step.dat"),
                modelout=here("20-three-step", "du3step_mplus", "c2_lca_du3step.inp") ,
                check=TRUE, run = TRUE, hashfilename = FALSE)
```

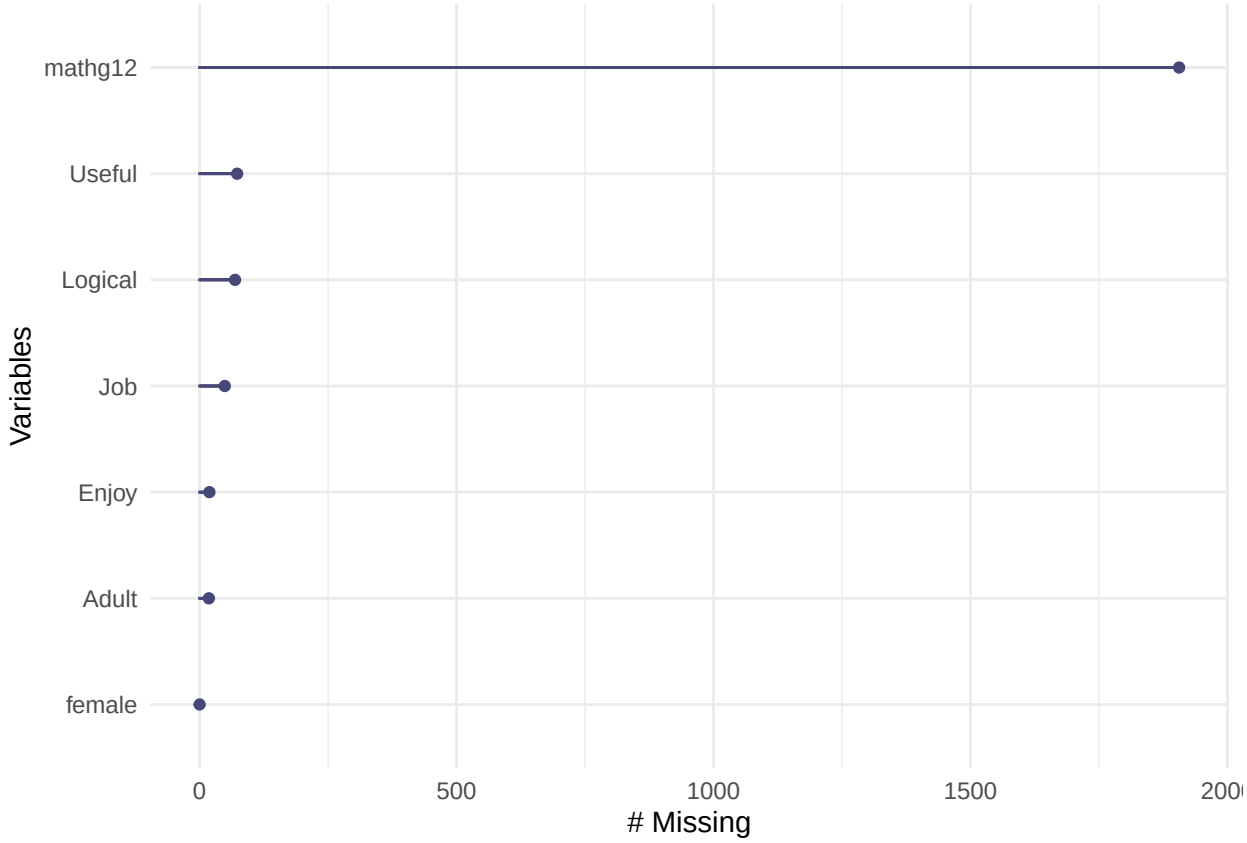
Application: Longitudinal Study of American Youth, Science Attitudes

Load data

```
lsay_data <- read_csv("https://garberadamc.github.io/project-site/data/lca_lsay_sci.csv",
  na = c("9999", "9999.00")) %>%
  clean_names() %>%
  dplyr::select(1:5, female, mathg12,
         Enjoy = ab39m, Useful = ab39t,
         Logical = ab39u, Job = ab39w, Adult = ab39x)
```

Use {naniar} to look at missing on covariates and distals

```
naniar::gg_miss_var(lsay_data)
```



Manual 3-step

- Adding covariates and distals to a mixture model

- Often called "*auxiliary variable integration*"

Step 1

```
step1  <- mplusObject(
  TITLE = "Step1 - 3step LSAY",
  VARIABLE =
  "categorical = Enjoy-Adult;
   usevar = Enjoy-Adult;

   classes = c(4);

   auxiliary =   ! list all potential covariates and distals here
   female        ! covariate
   mathg12;      ! distal math test score in 12th grade ",

  ANALYSIS =
   "estimator = mlr;
    type = mixture;
    starts = 500 100;",

  SAVEDATA =
   "File=3step_savedata.dat;
    Save=cprob;
    Missflag= 999;",

  OUTPUT = "sampstat residual tech11 tech14",

  PLOT =
    "type = plot3;
    series = Enjoy-Adult(*);",

  usevariables = colnames(lsay_data),
  rdata = lsay_data)

step1_fit <- mplusModeler(step1,
                          dataout=here("20-three-step", "3step_mplus", "Step1_3step_LSAY.dat"),
                          modelout=here("20-three-step", "3step_mplus", "Step1_3step_LSAY.inp") ,
                          check=TRUE, run = TRUE, hashfilename = FALSE)
```

Step 2

Extract logits for the classification probabilities for the most likely latent class

```
logit_cprobs <- as.data.frame(step1_fit[["results"]]
                                       [["class_counts"]]
                                       [["logitProbs.mostLikely"]])
```

Extract saved dataset which is part of the mplusObject "step1_10_fit"

```
savedata <- as.data.frame(step1_fit[["results"]]
                                    [["savedata"]])
```

Rename the column in savedata named "C" and change to "N"

```
colnames(savedata)[colnames(savedata)=="C"] <- "N"
```

Run step 2

```
step2  <- mplusObject(
  TITLE = "Step2 - 3step LSAY",

  VARIABLE =
 "nominal=N;
  USEVAR = n;
  missing are all (999);
  classes = c(4); ",

  ANALYSIS =
 "estimator = mlr;
  type = mixture;
  starts = 0;",

  MODEL =
    glue(
 "%C#1%
  [n#1@{logit_cprobs[1,1]}];
  [n#2@{logit_cprobs[1,2]}];
  [n#3@{logit_cprobs[1,3]}];

  %C#2%
  [n#1@{logit_cprobs[2,1]}];
  [n#2@{logit_cprobs[2,2]}];
  [n#3@{logit_cprobs[2,3]}];

  %C#3%
  [n#1@{logit_cprobs[3,1]}];
  [n#2@{logit_cprobs[3,2]}];
  [n#3@{logit_cprobs[3,3]}];

  %C#4%
  [n#1@{logit_cprobs[4,1]}];
  [n#2@{logit_cprobs[4,2]}];
  [n#3@{logit_cprobs[4,3]}];"),

  usevariables = colnames(savedata),
```

```
  rdata = savedata)

step2_fit <- mplusModeler(step2,
                          dataout=here("20-three-step", "3step_mplus", "Step2_3step_LSAY.dat"),
                          modelout=here("20-three-step", "3step_mplus", "Step2_3step_LSAY.inp"),
                          check=TRUE, run = TRUE, hashfilename = FALSE)
```

Step 3

Model with 1 covariate and 1 distal outcome

```
step3  <- mplusObject(
  TITLE = "Step3 - 3step LSAY",

  VARIABLE =
 "nominal=N;
  usevar = n;
  missing are all (999);
  classes = c(4);

  usevar = female mathg12;" ,

  ANALYSIS =
 "estimator = mlr;
  type = mixture;
  starts = 0;",

  MODEL =
  glue(
 " %OVERALL%

  C on female;      ! covariate as predictor of C

     %C#1%
  [n#1@{logit_cprobs[1,1]}];
  [n#2@{logit_cprobs[1,2]}];
  [n#3@{logit_cprobs[1,3]}];

  [mathg12](m1);    ! conditional distal mean
  mathg12;          ! conditional distal variance (freely estimated)

  %C#2%
  [n#1@{logit_cprobs[2,1]}];
  [n#2@{logit_cprobs[2,2]}];
  [n#3@{logit_cprobs[2,3]}];

  [mathg12](m2);
  mathg12;
```

```
  %C#3%
  [n#1@{logit_cprobs[3,1]}];
  [n#2@{logit_cprobs[3,2]}];
  [n#3@{logit_cprobs[3,3]}];

  [mathg12](m3);
  mathg12;

  %C#4%
  [n#1@{logit_cprobs[4,1]}];
  [n#2@{logit_cprobs[4,2]}];
  [n#3@{logit_cprobs[4,3]}];

  [mathg12](m4);
  mathg12; "),

  MODELCONSTRAINT =
   "New (diff12 diff13 diff23
    diff14 diff24 diff34);

    diff12 = m1-m2;  ! test pairwise distal mean differences
    diff13 = m1-m3;
    diff23 = m2-m3;
    diff14 = m1-m4;
    diff24 = m2-m4;
    diff34 = m3-m4;",

  MODELTEST = "     ! omnibus test of distal means
    m1=m2;
    m2=m3;
    m3=m4;",

  usevariables = colnames(savedata),
  rdata = savedata)

step3_fit <- mplusModeler(step3,
               dataout=here("20-three-step", "3step_mplus", "Step3_3step_LSAY.dat"),
               modelout=here("20-three-step", "3step_mplus", "Step3_3step_LSAY.inp"),
               check=TRUE, run = TRUE, hashfilename = FALSE)
```

End of manual 3-step

Model wtih latent categorical variable (C_k) as moderator

```
step3mod  <- mplusObject(
  TITLE = "Step3 - 3step LSAY",
```

```
 VARIABLE =
"nominal=N;
 usevar = n;
 missing are all (999);
 classes = c(4);

 usevar = female mathg12;" ,

 ANALYSIS =
"estimator = mlr;
 type = mixture;
 starts = 0;",

 MODEL =
 glue(
"!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 !!!!!DISTAL = mathg12, COVARIATE = female, MODERATOR = C!!!!!
 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 %OVERALL%
 mathg12 on female;
 mathg12;

    %C#1%
 [n#1@{logit_cprobs[1,1]}];
 [n#2@{logit_cprobs[1,2]}];
 [n#3@{logit_cprobs[1,3]}];

 mathg12 on female(s1);  ! conditional slope (class 1)
 [mathg12](m1);          ! conditional distal mean
 mathg12;                ! conditional distal variance (freely estimated)

 %C#2%
 [n#1@{logit_cprobs[2,1]}];
 [n#2@{logit_cprobs[2,2]}];
 [n#3@{logit_cprobs[2,3]}];

 mathg12 on female(s2);
 [mathg12](m2);
 mathg12;

 %C#3%
 [n#1@{logit_cprobs[3,1]}];
 [n#2@{logit_cprobs[3,2]}];
 [n#3@{logit_cprobs[3,3]}];

 mathg12 on female(s3);
 [mathg12](m3);
 mathg12;

 %C#4%
 [n#1@{logit_cprobs[4,1]}];
 [n#2@{logit_cprobs[4,2]}];
 [n#3@{logit_cprobs[4,3]}];
```

```
  mathg12 on female(s4);
  [mathg12](m4);
  mathg12; "),

  MODELCONSTRAINT =
   "New (slope12 slope13 slope23
    slope14 slope24 slope34);

    slope12 = s1-s2;  ! test pairwise slope differences
    slope13 = s1-s3;
    slope23 = s2-s3;
    slope14 = s1-s4;
    slope24 = s2-s4;
    slope34 = s3-s4;",

  MODELTEST = " ! can run only a single Omnibus test per model
    s1=s2;
    s2=s3;
    s3=s4;",

  usevariables = colnames(savedata),
  rdata = savedata)

step3mod_fit <- mplusModeler(step3mod,
                dataout=here("20-three-step", "3step_mplus", "Step3_moderation_LSAY.dat"),
                modelout=here("20-three-step", "3step_mplus", "Step3_moderation_LSAY.inp"),
                check=TRUE, run = TRUE, hashfilename = FALSE)
```

References

Drew A. Linzer, Jeffrey B. Lewis (2011). poLCA: An R Package for Polytomous Variable Latent Class Analysis. Journal of Statistical Software, 42(10), 1-29. URL http://www.jstatsoft.org/v42/i10/.

Hallquist, M. N., & Wiley, J. F. (2018). MplusAutomation: An R Package for Facilitating Large-Scale Latent Variable Analyses in Mplus. Structural equation modeling: a multidisciplinary journal, 25(4), 621-638.

Miller, J. D., Hoffer, T., Suchner, R., Brown, K., & Nelson, C. (1992). LSAY codebook. Northern Illinois University.

Muthén, B. O., Muthén, L. K., & Asparouhov, T. (2017). Regression and mediation analysis using Mplus. Los Angeles, CA: Muthén & Muthén.

Muthén, L.K. and Muthén, B.O. (1998-2017). Mplus User's Guide. Eighth Edition. Los Angeles, CA: Muthén & Muthén

R Core Team (2017). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/

Wickham et al., (2019). Welcome to the tidyverse. Journal of Open Source Software, 4(43), 1686, https://doi.org/10.21105/joss.01686