# Latent Profile Analysis Enumeration

IMMERSE Training Team

Updated: June 07, 2023

## Contents


**IMMERSE Project** **1**

Load packages . . . 2

Prepare Data . . . 2

Descriptive Statistics . . . 3

Enumeration . . . 3

Table of Fit . . . 5

Information Criteria Plot . . . 8

Compare models . . . 9

Latent Profile Plot . . . 12

References . . . 13


---

## IMMERSE Project

The Institute of Mixture Modeling for Equity-Oriented Researchers, Scholars, and Educators (IMMERSE) is an IES funded training grant (R305B220021) to support Education scholars in integrating mixture modeling into their research.

- Please visit our website to learn more and apply for the year-long fellowship.
- Follow us on Twitter!

How to reference this walkthrough: *This work was supported by the IMMERSE Project* (IES - 305B220021)

Visit our GitHub account to download the materials needed for this walkthrough.

---

*Example: PISA Student Data*

1. The first example closely follows the vignette used to demonstrate the tidyLPA package (Rosenberg, 2019).

- This model utilizes the `PISA` data collected in the U.S. in 2015. To learn more about this data see here.
- To access the 2015 US `PISA` data & documentation in R use the following code:

```
devtools::install_github("jrosen48/pisaUSA15")
library(pisaUSA15)
```

---

Latent Profile Models:

- `model 1` Class-invariant / Diagonal: Equal variances, and covariances fixed to 0
- `model 2` Class-varying / Diagonal: Free variances and covariances fixed to 0
- `model 3` Class-invariant / Non-Diagonal: Equal variances and equal covariances
- `model 4` Free variances, and equal covariances
- `model 5` Equal variances, and free covariances
- `model 6` Class Varying / Non-Diagonal: Free variances and free covariances

---

## Load packages

```
library(naniar)
library(tidyverse)
library(haven)
library(glue)
library(MplusAutomation)
library(here)
library(janitor)
library(gt)
library(tidyLPA)
library(pisaUSA15)
library(cowplot)
library(filesstrings)
here::i_am("lpa.Rmd")
```

---

## Prepare Data

```r
pisa <- pisaUSA15[1:500,] %>%
  dplyr::select(broad_interest, enjoyment, instrumental_mot, self_efficacy)
```

---

## Descriptive Statistics

```r
ds <- pisa %>%
  pivot_longer(broad_interest:self_efficacy, names_to = "variable") %>%
  group_by(variable) %>%
  summarise(mean = mean(value, na.rm = TRUE),
            sd = sd(value, na.rm = TRUE))

ds %>%
  gt () %>%
  tab_header(title = md("**Descriptive Summary**")) %>%
  cols_label(
    variable = "Variable",
    mean = md("M"),
    sd = md("SD")
  ) %>%
  fmt_number(c(2:3),
             decimals = 2) %>%
  cols_align(
    align = "center",
    columns = mean
  )
```

**Descriptive Summary**

| Variable | M | SD |
|---|---|---|
| broad_interest | 2.67 | 0.77 |
| enjoyment | 2.82 | 0.72 |
| instrumental_mot | 2.13 | 0.75 |
| self_efficacy | 2.12 | 0.64 |

---

## Enumeration

---

### `tidyLPA`

---

Enumerate using `estimate_profiles()`:

- Estimate models with classes $K = 1:4$
- Model has 4 continuous indicators
- Default variance-covariance specifications (model 1)
- Change `variances` and `covariances` to indicate the model you want to specify (see Vignette)

```
# Run LPA models
pisa %>%
    estimate_profiles(1:4,
                      package = "MplusAutomation",
                      ANALYSIS = "starts = 100, 20;",
                      variances = c("equal", "varying", "equal", "varying"),
                      covariances = c("zero", "zero", "equal", "equal"),
                      keepfiles = TRUE)

# Move files to folder
files <- list.files(here(), pattern = "^model")
file.move(files, here("tidyLPA"))
```

---

**`Mplus`**

---

Alternative method to `estimate_profiles()`: Run enumeration using `mplusObject` method

You can change the model specification for LPA using the syntax provided in lecture.

```

lpa_k14  <- lapply(1:4, function(k) {
  lpa_enum  <- mplusObject(

    TITLE = glue("Class {k}"),

    VARIABLE = glue(
    "usevar = broad_interest-self_efficacy;
     classes = c({k}); "),

  ANALYSIS =
   "estimator = mlr;
    type = mixture;
    starts = 100 20;",

  OUTPUT = "sampstat residual tech11 tech14;",

  usevariables = colnames(pisa),
  rdata = pisa)

lpa_enum_fit <- mplusModeler(lpa_enum,
                dataout=glue(here("enum_lpa", "lpa_pisa")),
                modelout=glue(here("enum_lpa", "c{k}_lpa_m1.inp")) ,
                check=TRUE, run = TRUE, hashfilename = FALSE)
})
```

### Model 1

```
lpa_m2_k14  <- lapply(1:4, function(k){

  MODEL <- lapply(1:k, function(i){

    glue("

    %c#{i}%
   broad_interest-self_efficacy;      ! variances are freely estimated

   ")
  })

  lpa_enum_m2  <- mplusObject(
    TITLE = glue("Class {k} - Model2"),

    VARIABLE = glue(
      "usevar = broad_interest-self_efficacy;
     classes = c({k});"),

    ANALYSIS =
      "estimator = mlr;
    type = mixture;
    starts = 100 20;",

    MODEL = glue("{MODEL[1:k]}"),


    OUTPUT = "sampstat residual tech11 tech14;",

    usevariables = colnames(pisa),
    rdata = pisa)

  lpa_m2_fit <- mplusModeler(lpa_enum_m2,
                             dataout = here("enum_lpa", "lpa_pisa"),
                             modelout = glue(here("enum_lpa","c{k}_lpa_m2.inp")),
                             check = TRUE, run = TRUE, hashfilename = FALSE)
})
```

### Model 2

---

## Table of Fit

APA formatted model fit table with additional fit indices

Extract data:

```
output_pisa <- readModels(here("tidyLPA"), quiet = TRUE)

enum_extract <- LatexSummaryTable(
  output_pisa,
  keepCols = c(
    "Title",
    "Parameters",
    "LL",
    "BIC",
    "aBIC",
    "BLRT_PValue",
    "Observations"
  )
)

allFit <- enum_extract %>%
  mutate(CAIC = -2 * LL + Parameters * (log(Observations) + 1)) %>%
  mutate(AWE = -2 * LL + 2 * Parameters * (log(Observations) + 1.5)) %>%
  separate(Title, c("Model", "Class"), sep = "with") %>%
  mutate(SIC = -.5 * BIC) %>%
  drop_na(SIC) %>%
  group_by(Model) %>%
  mutate(expSIC = exp(SIC - max(SIC))) %>%
  mutate(BF = exp(SIC - lead(SIC))) %>%
  mutate(cmPk = expSIC / sum(expSIC)) %>%
  ungroup() %>%
  unite(Title, c("Model", "Class"), sep = "with", remove = TRUE) %>%
  dplyr::select(1:5, 8:9, 6, 12, 13) %>%
  mutate(Title = str_to_title(Title)) %>%
  arrange(Title)
```

Create table:

```
allFit %>%
  gt() %>%
  tab_header(title = md("**Model Fit Summary Table**")) %>%
  cols_label(
    Title = "Classes",
    Parameters = md("Par"),
    LL = md("*LL*"),
    BLRT_PValue = "BLRT",
    BF = md("BF"),
    cmPk = md("*cmPk*")
  ) %>%
  tab_footnote(
    footnote = md(
      "*Note.* Par = Parameters; *LL* = model log likelihood;
BIC = Bayesian information criterion;
aBIC = sample size adjusted BIC; CAIC = consistent Akaike information criterion;
AWE = approximate weight of evidence criterion;
BLRT = bootstrapped likelihood ratio test p-value;
VLMR = Vuong-Lo-Mendell-Rubin adjusted likelihood ratio test p-value;
*cmPk* = approximate correct model probability."
```

```
    ),
locations = cells_title()
  )  %>%
  tab_options(column_labels.font.weight = "bold") %>%
  fmt_number(
    9,
    decimals = 2,
    drop_trailing_zeros = TRUE,
    suffixing = TRUE
  ) %>%
  fmt_number(c(3:8, 10),
             decimals = 0) %>%
  sub_missing(1:10,
              missing_text = "--") %>%
  fmt(
    c(8, 10),
    fns = function(x)
      ifelse(x < 0.001, "<0.001",
             scales::number(x, accuracy = 0.01))
  ) %>%
  fmt(
    9,
    fns = function (x)
      ifelse(x > 100, ">100",
             scales::number(x, accuracy = .1))
  ) %>%
  tab_row_group(
    label = "Model 1",
    rows = c(1:4)) %>%
  tab_row_group(
    label = "Model 2",
    rows =c(5:8)) %>%
  tab_row_group(
    label = "Model 3",
    rows = c(9:12)) %>%
  tab_row_group(
    label = "Model 4",
    rows = c(13:16)) %>%
  row_group_order(
    groups = c("Model 1","Model 2", "Model 3", "Model 4")
  )
```

**Model Fit Summary Table**[1]

| Classes | Par | *LL* | BIC | aBIC | CAIC | AWE | BLRT | BF | *cmPk* |
|---|---|---|---|---|---|---|---|---|---|
| Model 1 | | | | | | | | | |
| Model 1 With 1 Classes | 8 | $-2,089$ | $4,227$ | $4,201$ | $4,235$ | $4,300$ | – | 0.0 | <0.001 |
| Model 1 With 2 Classes | 13 | $-1,997$ | $4,074$ | $4,032$ | $4,087$ | $4,193$ | <0.001 | 0.0 | <0.001 |
| Model 1 With 3 Classes | 18 | $-1,953$ | $4,017$ | $3,960$ | $4,035$ | $4,183$ | <0.001 | 0.0 | <0.001 |
| Model 1 With 4 Classes | 23 | $-1,889$ | $3,921$ | $3,848$ | $3,944$ | $4,133$ | <0.001 | – | 1.00 |
| Model 2 | | | | | | | | | |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Model 2 With 1 Classes | 8 | $-2,089$ | $4,227$ | $4,201$ | $4,235$ | $4,300$ | – | 0.0 | <0.001 |
| Model 2 With 2 Classes | 17 | $-1,989$ | $4,083$ | $4,029$ | $4,100$ | $4,239$ | <0.001 | 0.0 | <0.001 |
| Model 2 With 3 Classes | 26 | $-1,878$ | $3,917$ | $3,834$ | $3,943$ | $4,156$ | <0.001 | >100 | 0.99 |
| Model 2 With 4 Classes | 35 | $-1,855$ | $3,927$ | $3,816$ | $3,962$ | $4,249$ | <0.001 | – | 0.01 |
| Model 3 | | | | | | | | | |
| Model 3 With 1 Classes | 14 | $-1,968$ | $4,023$ | $3,979$ | $4,037$ | $4,152$ | – | 0.1 | <0.001 |
| Model 3 With 2 Classes | 19 | $-1,950$ | $4,018$ | $3,958$ | $4,037$ | $4,192$ | <0.001 | 0.0 | 0.01 |
| Model 3 With 3 Classes | 24 | $-1,930$ | $4,009$ | $3,933$ | $4,033$ | $4,230$ | <0.001 | 1.9 | 0.65 |
| Model 3 With 4 Classes | 29 | $-1,916$ | $4,011$ | $3,919$ | $4,040$ | $4,277$ | <0.001 | – | 0.34 |
| Model 4 | | | | | | | | | |
| Model 4 With 1 Classes | 14 | $-1,968$ | $4,023$ | $3,979$ | $4,037$ | $4,152$ | – | 0.0 | <0.001 |
| Model 4 With 2 Classes | 23 | $-1,931$ | $4,004$ | $3,931$ | $4,027$ | $4,216$ | <0.001 | 0.0 | <0.001 |
| Model 4 With 3 Classes | 32 | $-1,859$ | $3,917$ | $3,815$ | $3,949$ | $4,211$ | <0.001 | >100 | 1.00 |
| Model 4 With 4 Classes | 41 | $-1,883$ | $4,021$ | $3,890$ | $4,062$ | $4,397$ | 0.01 | – | <0.001 |

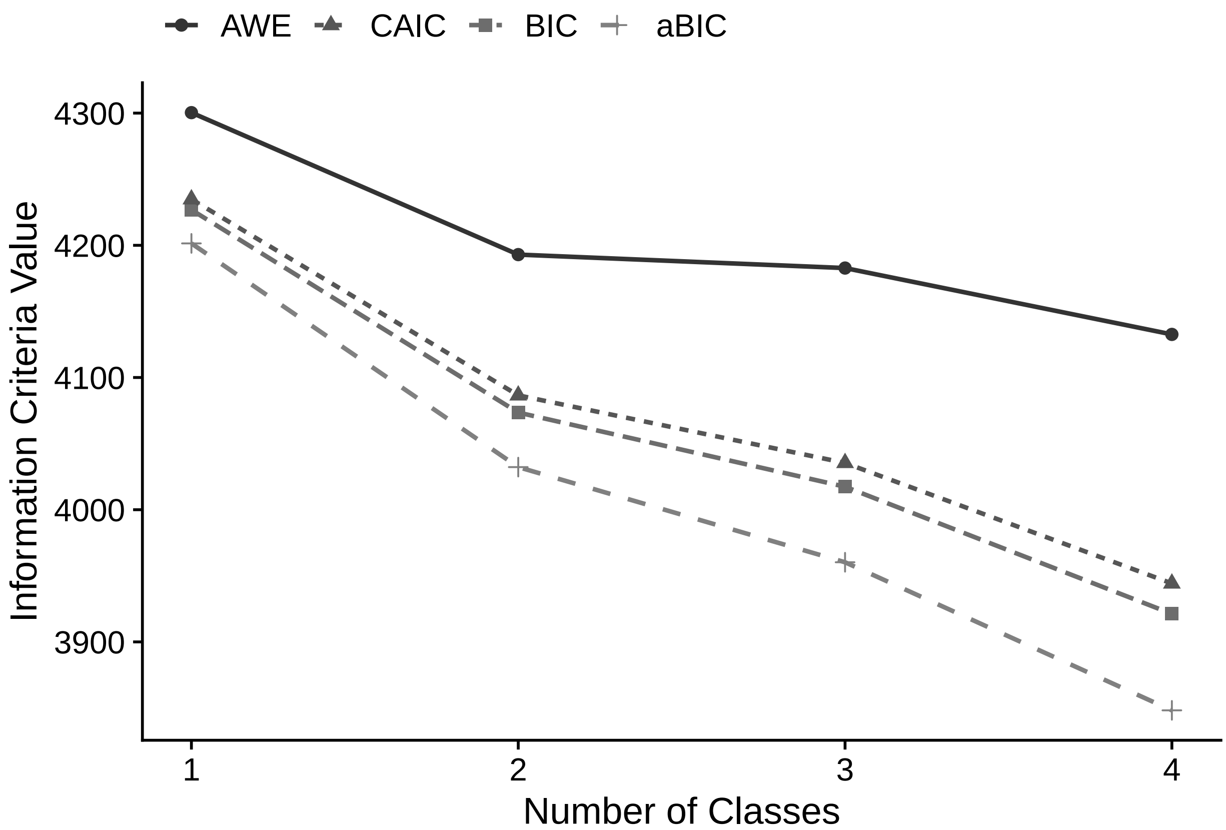
[1] *Note.* Par = Parameters; *LL* = model log likelihood; BIC = Bayesian information criterion; aBIC = sample size adjusted BIC; CAIC = consistent Akaike information criterion; AWE = approximate weight of evidence criterion; BLRT = bootstrapped likelihood ratio test p-value; VLMR = Vuong-Lo-Mendell-Rubin adjusted likelihood ratio test p-value; *cmPk* = approximate correct model probability.

---

## Information Criteria Plot

Plot information criteria

```
allFit %>%
  filter(grepl("Model 1", Title)) %>%
  dplyr::select(2:7) %>%
  rowid_to_column() %>%
  pivot_longer(`BIC`:`AWE`,
               names_to = "Index",
               values_to = "ic_value") %>%
mutate(Index = factor(Index,
                      levels = c ("AWE", "CAIC", "BIC", "aBIC"))) %>%
  ggplot(aes(x = rowid, y = ic_value,
             color = Index, shape = Index,
             group = Index, lty = Index)) +
  geom_point(size = 2.0) + geom_line(size = .8) +
  scale_x_continuous(breaks = 1:6) +
  scale_colour_grey(end = .5) +
  theme_cowplot() +
  labs(x = "Number of Classes", y = "Information Criteria Value") +
  theme(legend.title = element_blank(),
        legend.position = "top")
```




---

### Compare models

```r
# MplusAutomation Method using `compareModels`

parallelModels <- readModels(here("tidyLPA"))

compareModels(parallelModels[["model_3_class_2.out"]],
  parallelModels[["model_4_class_2.out"]], diffTest = TRUE)
```

```
##
## ==============
##
## Mplus model comparison
## ----------------------
##
## ------
## Model 1:  C:/Users/dinan/Box/IES_IMMERSE/Training Materials/lpa_enum/tidyLPA/model_3_class_2.out
## Model 2:  C:/Users/dinan/Box/IES_IMMERSE/Training Materials/lpa_enum/tidyLPA/model_4_class_2.out
## ------
##
## Model Summary Comparison
```

```
## ------------------------
##
##              m1                     m2
## Title        model 3 with 2 classes model 4 with 2 classes
## Observations 488                    488
## Estimator    MLR                    MLR
## Parameters   19                     23
## LL           -1950.111              -1930.959
## AIC          3938.222               3907.919
## BIC          4017.838               4004.296
##
##   MLR Chi-Square Difference Test for Nested Models Based on Loglikelihood
##   -----------------------------------------------------------------------
##
##   Difference Test Scaling Correction:  0.738925
##   Chi-square difference:  51.8375
##   Diff degrees of freedom:  4
##   P-value:  0
##
##   Note: The chi-square difference test assumes that these models are nested.
##   It is up to you to verify this assumption.
##
##   MLR Chi-Square Difference test for nested models
##   --------------------------------------------
##
##   Difference Test Scaling Correction:
##   Chi-square difference:
##   Diff degrees of freedom:
##   P-value:
##
## Note: The chi-square difference test assumes that these models are nested.
##   It is up to you to verify this assumption.
##
## =========
##
## Model parameter comparison
## --------------------------
##   Parameters present in both models
## =========
##
##   Approximately equal in both models (param. est. diff <= 1e-04)
##   ----------------------------------------------
## None
##
##
##   Parameter estimates that differ between models (param. est. diff > 1e-04)
##   ----------------------------------------------
##    paramHeader      param                 LatentClass m1_est m2_est . m1_se
##  BROAD_IN.WITH  ENJOYMENT                          1  0.263  0.201 | 0.030
##  BROAD_IN.WITH  ENJOYMENT                          2  0.263  0.201 | 0.030
##  BROAD_IN.WITH INSTRUMENT                          1 -0.133 -0.096 | 0.030
##  BROAD_IN.WITH INSTRUMENT                          2 -0.133 -0.096 | 0.030
##  BROAD_IN.WITH SELF_EFFIC                          1 -0.091 -0.078 | 0.027
##  BROAD_IN.WITH SELF_EFFIC                          2 -0.091 -0.078 | 0.027
```

```
##  ENJOYMEN.WITH INSTRUMENT                            1 -0.198 -0.140 | 0.030
##  ENJOYMEN.WITH INSTRUMENT                            2 -0.198 -0.140 | 0.030
##  ENJOYMEN.WITH SELF_EFFIC                            1 -0.139 -0.112 | 0.023
##  ENJOYMEN.WITH SELF_EFFIC                            2 -0.139 -0.112 | 0.023
##  INSTRUME.WITH SELF_EFFIC                            1  0.117  0.088 | 0.023
##  INSTRUME.WITH SELF_EFFIC                            2  0.117  0.088 | 0.023
##          Means BROAD_INTE                            1  2.645  2.790 | 0.036
##          Means BROAD_INTE                            2  3.221  2.406 | 0.270
##          Means       C1#1 Categorical.Latent.Variables  3.317  0.739 | 0.366
##          Means  ENJOYMENT                            1  2.805  2.982 | 0.033
##          Means  ENJOYMENT                            2  3.272  2.485 | 0.261
##          Means INSTRUMENT                            1  2.070  1.983 | 0.035
##          Means INSTRUMENT                            2  3.752  2.435 | 0.098
##          Means SELF_EFFIC                            1  2.138  2.065 | 0.030
##          Means SELF_EFFIC                            2  1.760  2.249 | 0.184
##      Variances BROAD_INTE                            1  0.584  0.410 | 0.038
##      Variances BROAD_INTE                            2  0.584  0.858 | 0.038
##      Variances  ENJOYMENT                            1  0.507  0.314 | 0.035
##      Variances  ENJOYMENT                            2  0.507  0.730 | 0.035
##      Variances INSTRUMENT                            1  0.464  0.344 | 0.037
##      Variances INSTRUMENT                            2  0.464  0.910 | 0.037
##      Variances SELF_EFFIC                            1  0.409  0.347 | 0.027
##      Variances SELF_EFFIC                            2  0.409  0.528 | 0.027
##  m2_se . m1_est_se m2_est_se . m1_pval m2_pval
##  0.033 |     8.836     6.174 |   0.000   0.000
##  0.033 |     8.836     6.174 |   0.000   0.000
##  0.031 |    -4.504    -3.077 |   0.000   0.002
##  0.031 |    -4.504    -3.077 |   0.000   0.002
##  0.028 |    -3.406    -2.831 |   0.001   0.005
##  0.028 |    -3.406    -2.831 |   0.001   0.005
##  0.024 |    -6.685    -5.750 |   0.000   0.000
##  0.024 |    -6.685    -5.750 |   0.000   0.000
##  0.024 |    -5.960    -4.577 |   0.000   0.000
##  0.024 |    -5.960    -4.577 |   0.000   0.000
##  0.025 |     5.108     3.557 |   0.000   0.000
##  0.025 |     5.108     3.557 |   0.000   0.000
##  0.060 |    74.314    46.719 |   0.000   0.000
##  0.112 |    11.934    21.469 |   0.000   0.000
##  0.281 |     9.058     2.630 |   0.000   0.009
##  0.044 |    84.651    68.113 |   0.000   0.000
##  0.116 |    12.558    21.453 |   0.000   0.000
##  0.045 |    58.820    44.495 |   0.000   0.000
##  0.101 |    38.149    24.215 |   0.000   0.000
##  0.057 |    71.163    36.164 |   0.000   0.000
##  0.109 |     9.587    20.656 |   0.000   0.000
##  0.057 |    15.492     7.149 |   0.000   0.000
##  0.119 |    15.492     7.204 |   0.000   0.000
##  0.032 |    14.381     9.772 |   0.000   0.000
##  0.069 |    14.381    10.538 |   0.000   0.000
##  0.033 |    12.422    10.567 |   0.000   0.000
##  0.100 |    12.422     9.061 |   0.000   0.000
##  0.051 |    14.936     6.789 |   0.000   0.000
##  0.076 |    14.936     6.921 |   0.000   0.000
##
```

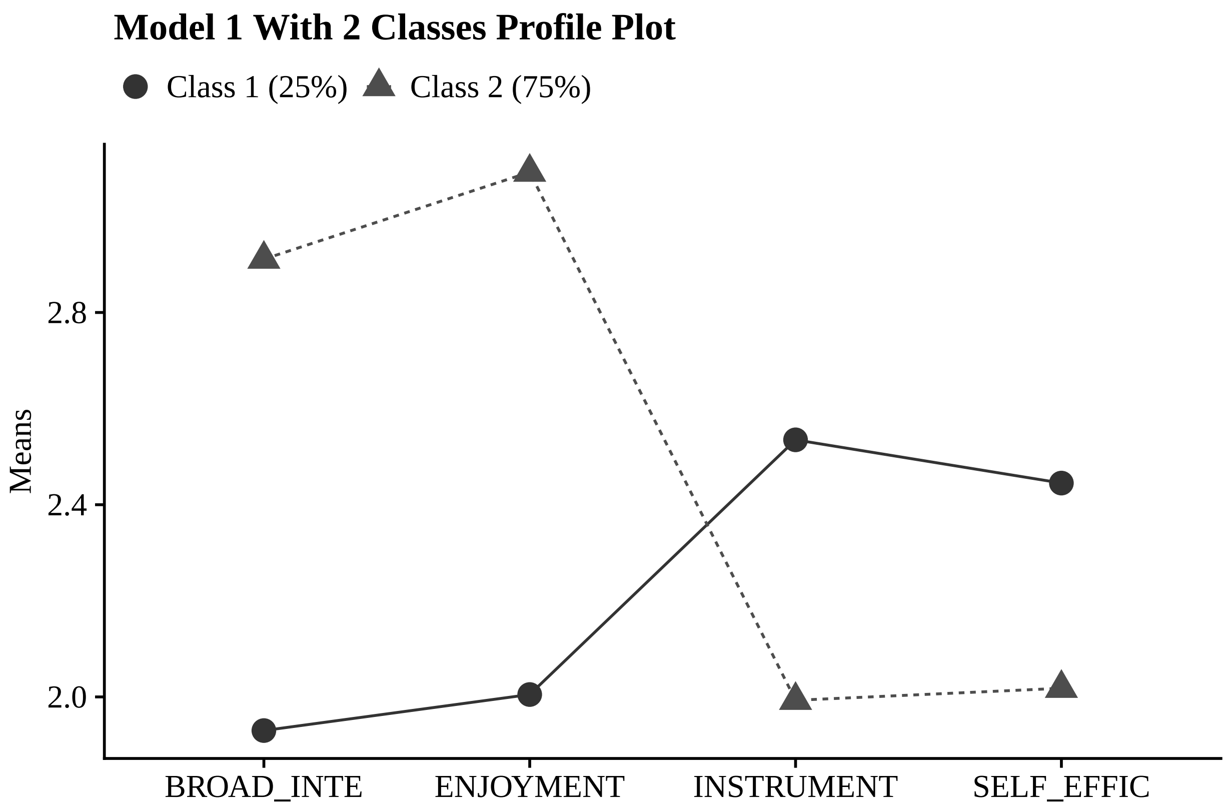
```
##
##   P-values that differ between models (p-value diff > 1e-04)
##   -----------------------------------
##    paramHeader      param                  LatentClass m1_est m2_est . m1_se
##  BROAD_IN.WITH INSTRUMENT                            1 -0.133 -0.096 | 0.030
##  BROAD_IN.WITH INSTRUMENT                            2 -0.133 -0.096 | 0.030
##  BROAD_IN.WITH SELF_EFFIC                            1 -0.091 -0.078 | 0.027
##  BROAD_IN.WITH SELF_EFFIC                            2 -0.091 -0.078 | 0.027
##          Means       C1#1 Categorical.Latent.Variables  3.317  0.739 | 0.366
##  m2_se . m1_est_se m2_est_se . m1_pval m2_pval
##  0.031 |    -4.504    -3.077 |   0.000   0.002
##  0.031 |    -4.504    -3.077 |   0.000   0.002
##  0.028 |    -3.406    -2.831 |   0.001   0.005
##  0.028 |    -3.406    -2.831 |   0.001   0.005
##  0.281 |     9.058     2.630 |   0.000   0.009
##
##
##   Parameters unique to model 1: 0
##   -----------------------------
##
##   None
##
##
##   Parameters unique to model 2: 0
##   -----------------------------
##
##  None
##
##
## ==============
```

---

### Latent Profile Plot

```r
source("plot_lpa_function.txt")

plot_lpa_function(model_name = output_pisa$model_1_class_2.out)
```




save figure

```
ggsave(here("figures", "C4_LPA_Plot.png"), dpi="retina", height=5, width=7, units="in")
```

---

## References


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

Rosenberg, J. M., van Lissa, C. J., Beymer, P. N., Anderson, D. J., Schell, M. J. & Schmidt, J. A. (2019). tidyLPA: Easily carry out Latent Profile Analysis (LPA) using open-source or commercial software [R package]. https://data-edu.github.io/tidyLPA/

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

UC SANTA BARBARA