|  |
| --- |
| LATENT CLASS ANALYSIS  Abstract  This article is based on a survey conducted for a student project at the University of Catania. The survey is mainly focused on investigating winery market trends and customer preferences for the Etna Wines. It contains twenty-three 23 questions about the drinking preferences, expertise, experiences, buying experiences, frequencies, and some demographic and personal information of participants. This article centered on Latent class analysis of buying experience and frequencies of the contestants. The associated data is classified into three different conceptual classes to find the potential drinkers and customers for the Etna Wines using the SAS data analysis tool. |



INTRODUCTION

The paper includes a broad overview of latent class analysis followed by an application using PROC LCA, (Lanza ST, Lemmon D, Schafer JL, Collins LM. 2006). The procedure is designed to work within SAS and is available at https://methodology.psu.edu/. For a comprehensive theoretical discussion of latent class analysis, see Collins and Lanza, (2013). The application is a “step-by-step” demonstration of data preparation, baseline model selection and identification, and extensions of LCA such as multiple-group LCA and LCA with covariates. Other options including the creation of output data sets and utilization of built-in SAS macros to prepare diagnostic plots are also covered.

The analysis of this application uses data collected from a survey conducted for a student project at the University of Catania. The purpose of the survey is to collect data on alcohol consumption to investigate the winery market trends and customer preferences. It contains twenty-three 23 questions about the drinking preferences, expertise, experiences, buying experiences, frequencies, and some demographic and personal information of participants. This article centered on Latent class analysis of buying experience and frequencies of the contestants. The associated data is classified into three different conceptual classes to find the potential drinkers and customers for the Etna Wines using the SAS data analysis tool.

Three analytic goals are addressed:

1. What patterns of underlying wine buying behaviors exist, can latent class analysis help explain those patterns, and, if so, what are the types and prevalence?

2. Is latent class measurement invariant across gender?

3. Does student groups and professional groups have differences in the behavior of drinking wine?

LATENT CLASS ANALYSIS

Latent class analysis is a statistical method used to identify unobserved or latent classes of individuals from observed responses to categorical variables (Goodman, 1974). It is analogous to factor analysis which is commonly used to identify latent classes for a set of continuous variables (Gorsuch, R. L.,1974). This technique offers a method for defining and analyzing unobserved classes and allows the analyst to make sense of a large number of possible combinations of responses from manifest variables. Two extensions of latent class analysis are multiple-group LCA and LCA with covariates. Multiple-group LCA permits class membership and item-response probabilities to vary across a group of interest where measurement invariance across groups can be tested. LCA with covariates extends the LCA model by including predictors (categorical or continuous) of class membership. For a more detailed discussion of LCA and other extensions, see Collins and Lanza (2013).

**PROC LCA**

The LCA procedure software and associated products including installation instructions, user documentation, analysis applications, SAS macros, recommended readings, and advanced extensions to PROC LCA can be downloaded from https://methodology.psu.edu/downloads/proclcalta. PROC LCA is organized much like a production SAS procedure and is easy to use and code. For more on required and optional procedure statements, see the documentation. Note that since PROC LCA is considered a production procedure in SAS v9.4, use of ODS destinations such as ODS HTML, ODS RTF, and ODS PDF are not available, therefore, list output from PROC LCA is used in this paper.

**DATA PREPARATION AND DESCRIPTIVE ANALYSIS**

Prior to the use of PROC LCA, data preparation consisting of sample refinement, variable construction, and descriptive analysis of key variables was performed.

The variables that explain the identifying the potential customers or consumers of the wine are given below.

1. BUYING\_EXPERIENCE
2. WINE\_BOTTLES

As these variables have categorical options as below.

1. BUYING\_EXPERIENCE
2. **1-2 times per month**
3. **3-4 times per month**
4. **5-6 times per month**
5. **7+ times per month**
6. **Never**
7. WINE\_BOTTLES
8. **1-3 bottles**
9. **10-12 bottles**
10. **12+ bottles**
11. **4-6 bottles**
12. **7-9 bottles**
13. **Less than 1 bottle**
14. **NA/Never**

In the second step the answers to these questions are converted into the following columns with the binary options 1 and 2. 1 for No and 2 for Yes.

| **Column Name** |
| --- |
| BUYING\_EXPERIENCE\_1\_2\_times\_per |
| BUYING\_EXPERIENCE\_3\_4\_times\_per |
| BUYING\_EXPERIENCE\_5\_6\_times\_per |
| BUYING\_EXPERIENCE\_7\_\_times\_per\_m |
| BUYING\_EXPERIENCE\_Never |
| WINE\_BOTTLES\_1\_3\_bottles |
| WINE\_BOTTLES\_10\_12\_bottles |
| WINE\_BOTTLES\_12\_\_bottles |
| WINE\_BOTTLES\_4\_6\_bottles |
| WINE\_BOTTLES\_7\_9\_bottles |
| WINE\_BOTTLES\_Less\_than\_1\_bottle |
| WINE\_BOTTLES\_NA |

The codes to create the above columns is given below.

**/\* define a macro to create dummy variables \*/**

**%macro DummyVars(DSIn, /\* the name of the input data set \*/**

**VarList, /\* the names of the categorical variables \*/**

**DSOut); /\* the name of the output data set \*/**

**/\* 1. add a fake response variable \*/**

**data AddFakeY / view=AddFakeY;**

**set &DSIn;**

**\_Y = 0; /\* add a fake response variable \*/**

**run;**

**/\* 2. Create the design matrix. Include the original variables, if desired \*/**

**proc glmselect data=AddFakeY NOPRINT outdesign(addinputvars)=&DSOut(drop=\_Y);**

**class &VarList;**

**model \_Y = &VarList / noint selection=none;**

**run;**

**%mend;**

**/\* test macro on the Age and Sex variables of the Sashelp.Class data \*/**

**%*DummyVars*(CONVERTED\_NUMERIC\_TO\_CHAR, BUYING\_EXPERIENCE WINE\_BOTTLES, DATA\_SET\_LATENT\_CLASS);**

**proc print data=DATA\_SET\_LATENT\_CLASS noobs;**

**VAR &\_GLSMod;**

**run;**

In the next step, these columns are given the labels from Item1 to Item 12. The code to give them these labels is given below:

**proc sql outobs=12;**

**select name**

**into :LATENT\_CLASS\_ITEMS separated by " "**

**from dictionary.columns**

**where libname = "WORK"**

**and memname = "DATA\_SET\_LATENT\_CLASS";**

**quit;**

**/\*create new dataset with missing values replaced by zero\*/**

**data DATA\_SET\_LATENT\_CLASS\_NEW;**

**set DATA\_SET\_LATENT\_CLASS;**

**array variablesOfInterest \_numeric\_;**

**do over variablesOfInterest;**

**if variablesOfInterest=1 then variablesOfInterest=2;**

**else if variablesOfInterest=0 then variablesOfInterest=1;**

**end;**

**run;**

**/\*view new dataset\*/**

**/\*proc print data=DATA\_SET\_LATENT\_CLASS\_NEW;\*/**

**data DATA\_SET\_LATENT\_CLASS\_NEW;**

**set DATA\_SET\_LC\_RENAMED;**

**rename BUYING\_EXPERIENCE\_1\_2\_times\_per = Item1**

**BUYING\_EXPERIENCE\_3\_4\_times\_per = Item2**

**BUYING\_EXPERIENCE\_5\_6\_times\_per = Item3**

**BUYING\_EXPERIENCE\_7\_\_times\_per\_m = Item4**

**BUYING\_EXPERIENCE\_Never = Item5**

**WINE\_BOTTLES\_1\_3\_bottles = Item6**

**WINE\_BOTTLES\_10\_12\_bottles = Item7**

**WINE\_BOTTLES\_12\_\_bottles = Item8**

**WINE\_BOTTLES\_4\_6\_bottles = Item9**

**WINE\_BOTTLES\_7\_9\_bottles = Item10**

**WINE\_BOTTLES\_Less\_than\_1\_bottle = Item11**

**WINE\_BOTTLES\_NA = Item12;**

**RUN;**

**The Frequency Table:**

| **BUYING\_EXPERIENCE** | **Frequency** | **Percent** | **Cumulative Frequency** | **Cumulative Percent** |
| --- | --- | --- | --- | --- |
| **1-2 times per month** | 110 | 44.53 | 110 | 44.53 |
| **3-4 times per month** | 44 | 17.81 | 154 | 62.35 |
| **5-6 times per month** | 18 | 7.29 | 172 | 69.64 |
| **7+ times per month** | 20 | 8.10 | 192 | 77.73 |
| **Never** | 55 | 22.27 | 247 | 100.00 |

| **WINE\_BOTTLES** | **Frequency** | **Percent** | **Cumulative Frequency** | **Cumulative Percent** |
| --- | --- | --- | --- | --- |
| **1-3 bottles** | 104 | 42.11 | 104 | 42.11 |
| **10-12 bottles** | 10 | 4.05 | 114 | 46.15 |
| **12+ bottles** | 8 | 3.24 | 122 | 49.39 |
| **4-6 bottles** | 37 | 14.98 | 159 | 64.37 |
| **7-9 bottles** | 14 | 5.67 | 173 | 70.04 |
| **Less than 1 bottle** | 19 | 7.69 | 192 | 77.73 |
| **NA** | 55 | 22.27 | 247 | 100.00 |

The frequency table above for the BUYING\_EXPERIENCE attribute reveals some initial information regarding the model selection for the latent class analysis. According to the table, around 8.10 percent of respondents drink wine 7+ times per month and 7.29 percent of respondents drink wine 5-6 times per month. After combining these percentages we get 15.39 percent. So, we can consider enough amount of respondents who are frequent consumers of the wine and could be considered potential customers. While on the other side about 44.53 percent of people buy wines 1-2 times per month. Those buyers could be considered the least potential customers. In parallel to this, there are some buyers which lie in the middle 3 – 4 times per month. This pattern shows a clear synergy in order to segregate the data into 3 different latent classes.

Similarly, for the attribute number of the consumption of the WINE\_BOTTLES respondents which consumes 10-12 bottles,12+bottles, and 7-9 bottles are around 12.29 percent. Which could be considered as potential customers. In this attribute the same synergy is present we can divide that data into three different classes like the buying\_experience attribute. For example, 42.11 percent of respondents are considered and least potential consumers while 14.98 percent of consumers can be considered mid-level consumers.

The SAS code to generate these frequency tables is given below:

%let DSIn = CONVERTED\_NUMERIC\_TO\_CHAR; /\* name of input data set \*/

%let VarList = BUYING\_EXPERIENCE WINE\_BOTTLES; /\* name of categorical variables \*/

**proc** **freq** data=&DSIn;

tables &VarList;

**run**;

**Mean and Standard Deviation of the created Items:**

The mean and standard deviation of the answered labels and their associated variables is given below in the table with its SAS code in the separate section.

| **Variable** | **Label** | **Mean** | **Std Dev** |
| --- | --- | --- | --- |
| |  | | --- | | **Item1** | | **Item2** | | **Item3** | | **Item4** | | **Item5** | | **Item6** | | **Item7** | | **Item8** | | **Item9** | | **Item10** | | **Item11** | | **Item12** | | |  | | --- | | **BUYING\_EXPERIENCE 1-2 times per month** | | **BUYING\_EXPERIENCE 3-4 times per month** | | **BUYING\_EXPERIENCE 5-6 times per month** | | **BUYING\_EXPERIENCE 7+ times per month** | | **BUYING\_EXPERIENCE Never** | | **WINE\_BOTTLES 1-3 bottles** | | **WINE\_BOTTLES 10-12 bottles** | | **WINE\_BOTTLES 12+ bottles** | | **WINE\_BOTTLES 4-6 bottles** | | **WINE\_BOTTLES 7-9 bottles** | | **WINE\_BOTTLES Less than 1 bottle** | | **WINE\_BOTTLES NA** | | |  | | --- | | 1.4453441 | | 1.1781377 | | 1.0728745 | | 1.0809717 | | 1.2226721 | | 1.4210526 | | 1.0404858 | | 1.0323887 | | 1.1497976 | | 1.0566802 | | 1.0769231 | | 1.2226721 | | |  | | --- | | 0.4980129 | | 0.3834055 | | 0.2604582 | | 0.2733455 | | 0.4168847 | | 0.4947305 | | 0.1974959 | | 0.1773895 | | 0.3575975 | | 0.2317000 | | 0.2670104 | | 0.4168847 | |

**proc** **means** data = DATA\_SET\_LC\_RENAMED mean std;

var Item1 - item12;

**run**;

**BASELINE MODEL SELECTION**

To address our first research question: “What patterns of underlying alcohol behaviors exist, can latent class analysis help explain those patterns and, if so, what are the types and prevalence? PROC LCA is used repeatedly to analyze models with 2-5 classes. The goal is a selection of an optimal baseline model from the four LCA models tested. This process can be challenging given the interplay of different factors such as evaluation of model fit statistics, model identification, class membership probabilities, and interpretability of latent classes.

Initially, PROC LCA is executed six times using the user-defined macro code below. Each model uses 300 random starts and seven alcohol behavior variables (coded 1 or 2). The % wine\_proclca macro includes required and optional statements such as PROC LCA with ORIG\_WEIGHTS, OUTEST, OUTPOST statements along with WEIGHT, ID, NSTARTS, NCLASS, SEED, and RHO PRIOR statements. See the User’s Guide for more detail on syntax. The option outparam allows us to save the final parameter estimates for each item and each class and outpost allows us to save the posterior probabilities for each observation.

The PROC LCA statement includes use of the probability weight with the ORIG\_WEIGHTS option and also requests output data sets of parameter estimates (OUTEST with a macro variable that resolves to the number of classes) and posterior probabilities (OUTPOST with same macro variable resolution). Additional options define the id (ID) needed for future data set manipulation, number of classes (NCLASS), number of random starts (NSTARTS), number of CPU cores used to process the job (CORES), a seed value (SEED), a weight variable (WEIGHT), the alcohol behavior items and the corresponding number of categories (ITEMS, CATEGORIES), and a prior used in the calculation of the Rho values (RHO PRIOR):

**%macro wine\_proclca(nc);**

**proc lca data=DATA\_SET\_LC\_RENAMED orig\_weights OUTPARAM = test&nc outpost = lca1\_post&nc;**

**id id ;**

**nstarts 300 ;**

**nclass &nc ;**

**cores 4 ;**

**items Item1 Item2 Item3 Item4 Item5 Item6 Item7 Item8 Item9 Item10 Item11 Item12;**

**categories 2 2 2 2 2 2 2 2 2 2 2 2;**

**seed 1232 ;**

**rho prior=1;**

**run ;**

**%mend wine\_proclca;**

%***wine\_proclca***(**2**) ;

%***wine\_proclca***(**3**) ;

%***wine\_proclca***(**4**) ;

%***wine\_proclca***(**5**);

**2- Classes OUTPUT:**

The below sections includes the output of the two classes model for the winery data. The out highlighted in red color is the important parts in this proc lca analysis for the 2 latent classes.

**Data Summary, Model Information, and Fit Statistics (EM Algorithm)**

**Number of subjects in dataset: 247**

**Number of subjects in analysis: 247**

**Number of measurement items: 12**

**Response categories per item: 2 2 2 2 2 2 2 2 2 2 2 2**

**Number of groups in the data: 1**

**Number of latent classes: 2**

**NOTE: A data-derived prior was applied to the rho parameters to help**

**avoid parameter estimates on boundary values of zero and one.**

**Rho starting values were randomly generated (seed = 1232).**

**No parameter restrictions were specified (freely estimated).**

**Seed selected for best fitted model: 776037990**

**Percentage of seeds associated with best fitted model: 27.67%**

**The model converged in 6 iterations.**

**Maximum number of iterations: 5000**

**Convergence method: maximum absolute deviation (MAD)**

**Convergence criterion: 0.000001000**

**=============================================**

**Fit statistics:**

**=============================================**

**Log-likelihood: -902.19**

**G-squared: 766.87**

**AIC: 816.87**

**BIC: 904.61**

**CAIC: 929.61**

**Adjusted BIC: 825.36**

**Entropy: 1.00**

**Degrees of freedom: 4070**

**Parameter Estimates**

**Class membership probabilities: Gamma estimates (standard errors)**

**Class: 1 2**

**0.7773 0.2227**

**(0.0265) (0.0265)**

**Item response probabilities: Rho estimates (standard errors)**

**Response category 1:**

**Class: 1 2**

**Item1 : 0.4274 0.9960**

**(0.0357) (0.0085)**

**Item2 : 0.7710 0.9984**

**(0.0303) (0.0054)**

**Item3 : 0.9063 0.9993**

**(0.0210) (0.0034)**

**Item4 : 0.8959 0.9993**

**(0.0220) (0.0036)**

**Item5 : 0.9994 0.0070**

**(0.0017) (0.0112)**

**Item6 : 0.4586 0.9962**

**(0.0359) (0.0083)**

**Item7 : 0.9479 0.9996**

**(0.0160) (0.0026)**

**Item8 : 0.9584 0.9997**

**(0.0144) (0.0023)**

**Item9 : 0.8074 0.9987**

**(0.0284) (0.0049)**

**Item10 : 0.9271 0.9995**

**(0.0187) (0.0030)**

**Item11 : 0.9011 0.9993**

**(0.0215) (0.0035)**

**Item12 : 0.9994 0.0070**

**(0.0017) (0.0112)**

**Response category 2:**

**Class: 1 2**

**Item1 : 0.5726 0.0040**

**(0.0357) (0.0085)**

**Item2 : 0.2290 0.0016**

**(0.0303) (0.0054)**

**Item3 : 0.0937 0.0007**

**(0.0210) (0.0034)**

**Item4 : 0.1041 0.0007**

**(0.0220) (0.0036)**

**Item5 : 0.0006 0.9930**

**(0.0017) (0.0112)**

**Item6 : 0.5414 0.0038**

**(0.0359) (0.0083)**

**Item7 : 0.0521 0.0004**

**(0.0160) (0.0026)**

**Item8 : 0.0416 0.0003**

**(0.0144) (0.0023)**

**Item9 : 0.1926 0.0013**

**(0.0284) (0.0049)**

**Item10 : 0.0729 0.0005**

**(0.0187) (0.0030)**

**Item11 : 0.0989 0.0007**

**(0.0215) (0.0035)**

**Item12 : 0.0006 0.9930**

**(0.0017) (0.0112)**

**3-Classes Model:**

Data Summary, Model Information, and Fit Statistics (EM Algorithm)

Number of subjects in dataset: 247

Number of subjects in analysis: 247

Number of measurement items: 12

Response categories per item: 2 2 2 2 2 2 2 2 2 2 2 2

Number of groups in the data: 1

Number of latent classes: 3

NOTE: A data-derived prior was applied to the rho parameters to help

avoid parameter estimates on boundary values of zero and one.

Rho starting values were randomly generated (seed = 1232).

No parameter restrictions were specified (freely estimated).

Seed selected for best fitted model: 2106526603

Percentage of seeds associated with best fitted model: 35.67%

The model converged in 13 iterations.

Maximum number of iterations: 5000

Convergence method: maximum absolute deviation (MAD)

Convergence criterion: 0.000001000

=============================================

Fit statistics:

=============================================

Log-likelihood: -753.24

G-squared: 468.97

AIC: 544.97

BIC: 678.33

CAIC: 716.33

Adjusted BIC: 557.87

Entropy: 1.00

Degrees of freedom: 4057

Parameter Estimates

Class membership probabilities: Gamma estimates (standard errors)

Class: 1 2 3

0.2227 0.3324 0.4449

(0.0265) (0.0300) (0.0317)

Item response probabilities: Rho estimates (standard errors)

Response category 1:

Class: 1 2 3

Item1 : 0.9973 0.9968 0.0017

(0.0070) (0.0082) (0.0039)

Item2 : 0.9989 0.4656 0.9995

(0.0044) (0.0550) (0.0022)

Item3 : 0.9996 0.7814 0.9998

(0.0028) (0.0455) (0.0014)

Item4 : 0.9995 0.7571 0.9998

(0.0030) (0.0472) (0.0015)

Item5 : 0.0047 0.9991 0.9993

(0.0092) (0.0033) (0.0025)

Item6 : 0.9975 0.7435 0.2458

(0.0068) (0.0481) (0.0411)

Item7 : 0.9998 0.8905 0.9909

(0.0021) (0.0344) (0.0091)

Item8 : 0.9998 0.9028 0.9999

(0.0019) (0.0326) (0.0009)

Item9 : 0.9991 0.6347 0.9366

(0.0040) (0.0531) (0.0233)

Item10 : 0.9997 0.8418 0.9910

(0.0025) (0.0402) (0.0091)

Item11 : 0.9995 0.9875 0.8365

(0.0029) (0.0122) (0.0352)

Item12 : 0.0047 0.9991 0.9993

(0.0092) (0.0033) (0.0025)

Response category 2:

Class: 1 2 3

Item1 : 0.0027 0.0032 0.9983

(0.0070) (0.0082) (0.0039)

Item2 : 0.0011 0.5344 0.0005

(0.0044) (0.0550) (0.0022)

Item3 : 0.0004 0.2186 0.0002

(0.0028) (0.0455) (0.0014)

Item4 : 0.0005 0.2429 0.0002

(0.0030) (0.0472) (0.0015)

Item5 : 0.9953 0.0009 0.0007

(0.0092) (0.0033) (0.0025)

Item6 : 0.0025 0.2565 0.7542

(0.0068) (0.0481) (0.0411)

Item7 : 0.0002 0.1095 0.0091

(0.0021) (0.0344) (0.0091)

Item8 : 0.0002 0.0972 0.0001

(0.0019) (0.0326) (0.0009)

Item9 : 0.0009 0.3653 0.0634

(0.0040) (0.0531) (0.0233)

Item10 : 0.0003 0.1582 0.0090

(0.0025) (0.0402) (0.0091)

Item11 : 0.0005 0.0125 0.1635

(0.0029) (0.0122) (0.0352)

Item12 : 0.9953 0.0009 0.0007

(0.0092) (0.0033) (0.0025)

**4- Classes Model:**

Data Summary, Model Information, and Fit Statistics (EM Algorithm)

Number of subjects in dataset: 247

Number of subjects in analysis: 247

Number of measurement items: 12

Response categories per item: 2 2 2 2 2 2 2 2 2 2 2 2

Number of groups in the data: 1

Number of latent classes: 4

NOTE: A data-derived prior was applied to the rho parameters to help

avoid parameter estimates on boundary values of zero and one.

Rho starting values were randomly generated (seed = 1232).

No parameter restrictions were specified (freely estimated).

Seed selected for best fitted model: 473714953

Percentage of seeds associated with best fitted model: 23.00%

The model converged in 16 iterations.

Maximum number of iterations: 5000

Convergence method: maximum absolute deviation (MAD)

Convergence criterion: 0.000001000

=============================================

Fit statistics:

=============================================

Log-likelihood: -684.21

G-squared: 330.91

AIC: 432.91

BIC: 611.89

CAIC: 662.89

Adjusted BIC: 450.22

Entropy: 1.00

Degrees of freedom: 4044

1 2 3 4

0.4450 0.1781 0.1542 0.2227

(0.0317) (0.0244) (0.0230) (0.0265)

Item response probabilities: Rho estimates (standard errors)

Response category 1:

Class: 1 2 3 4

Item1 : 0.0013 0.9975 0.9949 0.9980

(0.0034) (0.0076) (0.0152) (0.0060)

Item2 : 0.9996 0.0047 0.9984 0.9992

(0.0019) (0.0103) (0.0073) (0.0038)

Item3 : 0.9998 0.9996 0.5301 0.9997

(0.0012) (0.0031) (0.0808) (0.0024)

Item4 : 0.9998 0.9995 0.4779 0.9996

(0.0013) (0.0032) (0.0809) (0.0026)

Item5 : 0.9995 0.9987 0.9985 0.0035

(0.0021) (0.0053) (0.0061) (0.0080)

Item6 : 0.2456 0.5681 0.9451 0.9981

(0.0411) (0.0745) (0.0368) (0.0059)

Item7 : 0.9911 0.9998 0.7644 0.9998

(0.0091) (0.0023) (0.0686) (0.0018)

Item8 : 0.9999 0.9773 0.8171 0.9999

(0.0008) (0.0224) (0.0625) (0.0016)

Item9 : 0.9363 0.4792 0.8160 0.9993

(0.0233) (0.0751) (0.0627) (0.0035)

Item10 : 0.9911 0.9773 0.6855 0.9997

(0.0092) (0.0225) (0.0751) (0.0022)

Item11 : 0.8365 0.9996 0.9734 0.9997

(0.0352) (0.0031) (0.0260) (0.0025)

Item12 : 0.9995 0.9987 0.9985 0.0035

(0.0021) (0.0053) (0.0061) (0.0080)

Response category 2:

Class: 1 2 3 4

Item1 : 0.9987 0.0025 0.0051 0.0020

(0.0034) (0.0076) (0.0152) (0.0060)

Item2 : 0.0004 0.9953 0.0016 0.0008

(0.0019) (0.0103) (0.0073) (0.0038)

Item3 : 0.0002 0.0004 0.4699 0.0003

(0.0012) (0.0031) (0.0808) (0.0024)

Item4 : 0.0002 0.0005 0.5221 0.0004

(0.0013) (0.0032) (0.0809) (0.0026)

Item5 : 0.0005 0.0013 0.0015 0.9965

(0.0021) (0.0053) (0.0061) (0.0080)

Item6 : 0.7544 0.4319 0.0549 0.0019

(0.0411) (0.0745) (0.0368) (0.0059)

Item7 : 0.0089 0.0002 0.2356 0.0002

(0.0091) (0.0023) (0.0686) (0.0018)

Item8 : 0.0001 0.0227 0.1829 0.0001

(0.0008) (0.0224) (0.0625) (0.0016)

Item9 : 0.0637 0.5208 0.1840 0.0007

(0.0233) (0.0751) (0.0627) (0.0035)

Item10 : 0.0089 0.0227 0.3145 0.0003

(0.0092) (0.0225) (0.0751) (0.0022)

Item11 : 0.1635 0.0004 0.0266 0.0003

(0.0352) (0.0031) (0.0260) (0.0025)

Item12 : 0.0005 0.0013 0.0015 0.9965

(0.0021) (0.0053) (0.0061) (0.0080)

**5-Classes Model:**

Data Summary, Model Information, and Fit Statistics (EM Algorithm)

Number of subjects in dataset: 247

Number of subjects in analysis: 247

Number of measurement items: 12

Response categories per item: 2 2 2 2 2 2 2 2 2 2 2 2

Number of groups in the data: 1

Number of latent classes: 5

NOTE: A data-derived prior was applied to the rho parameters to help

avoid parameter estimates on boundary values of zero and one.

Rho starting values were randomly generated (seed = 1232).

No parameter restrictions were specified (freely estimated).

Seed selected for best fitted model: 1137544060

Percentage of seeds associated with best fitted model: 10.67%

The model converged in 56 iterations.

Maximum number of iterations: 5000

Convergence method: maximum absolute deviation (MAD)

Convergence criterion: 0.000001000

=============================================

Fit statistics:

=============================================

Log-likelihood: -639.09

G-squared: 240.68

AIC: 368.68

BIC: 593.28

CAIC: 657.28

Adjusted BIC: 390.40

Entropy: 1.00

Degrees of freedom: 4031

Parameter Estimates

Class membership probabilities: Gamma estimates (standard errors)

Class: 1 2 3 4 5

0.1092 0.1541 0.3359 0.2227 0.1781

(0.0199) (0.0230) (0.0301) (0.0265) (0.0243)

Item response probabilities: Rho estimates (standard errors)

Response category 1:

Class: 1 2 3 4 5

Item1 : 0.0041 0.9965 0.0013 0.9984 0.9980

(0.0123) (0.0116) (0.0040) (0.0054) (0.0068)

Item2 : 0.9987 0.9988 0.9996 0.9994 0.0037

(0.0069) (0.0066) (0.0023) (0.0034) (0.0092)

Item3 : 0.9994 0.5291 0.9998 0.9997 0.9997

(0.0046) (0.0808) (0.0015) (0.0022) (0.0028)

Item4 : 0.9994 0.4768 0.9998 0.9997 0.9996

(0.0047) (0.0808) (0.0015) (0.0023) (0.0029)

Item5 : 0.9984 0.9988 0.9995 0.0028 0.9990

(0.0078) (0.0055) (0.0025) (0.0071) (0.0048)

Item6 : 0.9960 0.9455 0.0014 0.9985 0.5681

(0.0138) (0.0367) (0.0041) (0.0053) (0.0745)

Item7 : 0.9634 0.7642 0.9999 0.9999 0.9998

(0.0363) (0.0687) (0.0011) (0.0016) (0.0020)

Item8 : 0.9998 0.8168 0.9999 0.9999 0.9773

(0.0030) (0.0626) (0.0010) (0.0015) (0.0224)

Item9 : 0.7417 0.8160 0.9996 0.9995 0.4789

(0.0840) (0.0627) (0.0021) (0.0031) (0.0752)

Item10 : 0.9635 0.6853 0.9999 0.9998 0.9773

(0.0364) (0.0752) (0.0013) (0.0019) (0.0225)

Item11 : 0.3372 0.9735 0.9998 0.9997 0.9997

(0.0910) (0.0260) (0.0015) (0.0022) (0.0028)

Item12 : 0.9984 0.9988 0.9995 0.0028 0.9990

(0.0078) (0.0055) (0.0025) (0.0071) (0.0048)

Response category 2:

Class: 1 2 3 4 5

Item1 : 0.9959 0.0035 0.9987 0.0016 0.0020

(0.0123) (0.0116) (0.0040) (0.0054) (0.0068)

Item2 : 0.0013 0.0012 0.0004 0.0006 0.9963

(0.0069) (0.0066) (0.0023) (0.0034) (0.0092)

Item3 : 0.0006 0.4709 0.0002 0.0003 0.0003

(0.0046) (0.0808) (0.0015) (0.0022) (0.0028)

Item4 : 0.0006 0.5232 0.0002 0.0003 0.0004

(0.0047) (0.0808) (0.0015) (0.0023) (0.0029)

Item5 : 0.0016 0.0012 0.0005 0.9972 0.0010

(0.0078) (0.0055) (0.0025) (0.0071) (0.0048)

Item6 : 0.0040 0.0545 0.9986 0.0015 0.4319

(0.0138) (0.0367) (0.0041) (0.0053) (0.0745)

Item7 : 0.0366 0.2358 0.0001 0.0001 0.0002

(0.0363) (0.0687) (0.0011) (0.0016) (0.0020)

Item8 : 0.0002 0.1832 0.0001 0.0001 0.0227

(0.0030) (0.0626) (0.0010) (0.0015) (0.0224)

Item9 : 0.2583 0.1840 0.0004 0.0005 0.5211

(0.0840) (0.0627) (0.0021) (0.0031) (0.0752)

Item10 : 0.0365 0.3147 0.0001 0.0002 0.0227

(0.0364) (0.0752) (0.0013) (0.0019) (0.0225)

Item11 : 0.6628 0.0265 0.0002 0.0003 0.0003

(0.0910) (0.0260) (0.0015) (0.0022) (0.0028)

Item12 : 0.0016 0.0012 0.0005 0.9972 0.0010

(0.0078) (0.0055) (0.0025) (0.0071) (0.0048)

**Model Fit Comparison**

The following code uses a macro called %it to create a new variable called NCLASS with values of 2-5 to assign a class number to each output data set. Then the four output datasets are concatenated to produce a summary data set called all **lca1\_allfit\_alc**. Finally, PROC PRINT is used to produce a model fit comparison:

| **Number of Classes** | **LL** | **DF** | **G\_SQUARED** | **AIC** | **BIC** | **CAIC** | **ABIC** | **ENTROPY** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **2** | **-902.188673** | **4070** | **766.87424127** | **816.87424127** | **904.60894968** | **929.60894968** | **825.35921792** | **0.9999994154** |
| **3** | **-753.2361172** | **4057** | **468.96912975** | **544.96912975** | **678.32588654** | **716.32588654** | **557.86629426** | **0.9975464629** |
| **4** | **-684.2072103** | **4044** | **330.91131598** | **432.91131598** | **611.89012115** | **662.89012115** | **450.22066835** | **0.9983848819** |
| **5** | **-639.0921444** | **4031** | **240.68118413** | **368.68118413** | **593.28203767** | **657.28203767** | **390.40272436** | **0.9985460031** |

The above table provides a comparative summary of model fit statistics. For the AIC, BIC, CAIC, ABIC, and G2 statistics, lower values generally indicate better model fit while higher values on Entropy indicate better separation of latent classes. Here, the five class models appear to be the best fit since the AIC, BIC, CAIC, and ABIC are each the lowest for this model. Also, the G 2= 240.6 with 4031 degrees of freedom for the 5 class model shows a large similarity from the 4 class model with G 2 =330.91 and 4031 degrees of freedom. Entropy equals 0.998 which suggests good class interpretability and separation. On the contrary, when the 4 classes of model item response probabilities were deeply investigated then we found that it is more convenient to consider that model as the best fit in the winery data analysis. The below table with response category 2 represents

Response category 2:

Class: 1 2 3 4

Item1 : 0.9987 0.0025 0.0051 0.0020

(0.0034) (0.0076) (0.0152) (0.0060)

Item2 : 0.0004 0.9953 0.0016 0.0008

(0.0019) (0.0103) (0.0073) (0.0038)

Item3 : 0.0002 0.0004 0.4699 0.0003

(0.0012) (0.0031) (0.0808) (0.0024)

Item4 : 0.0002 0.0005 0.5221 0.0004

(0.0013) (0.0032) (0.0809) (0.0026)

Item5 : 0.0005 0.0013 0.0015 0.9965

(0.0021) (0.0053) (0.0061) (0.0080)

Item6 : 0.7544 0.4319 0.0549 0.0019

(0.0411) (0.0745) (0.0368) (0.0059)

Item7 : 0.0089 0.0002 0.2356 0.0002

(0.0091) (0.0023) (0.0686) (0.0018)

Item8 : 0.0001 0.0227 0.1829 0.0001

(0.0008) (0.0224) (0.0625) (0.0016)

Item9 : 0.0637 0.5208 0.1840 0.0007

(0.0233) (0.0751) (0.0627) (0.0035)

Item10 : 0.0089 0.0227 0.3145 0.0003

(0.0092) (0.0225) (0.0751) (0.0022)

Item11 : 0.1635 0.0004 0.0266 0.0003

(0.0352) (0.0031) (0.0260) (0.0025)

Item12 : 0.0005 0.0013 0.0015 0.9965

(0.0021) (0.0053) (0.0061) (0.0080)

As a preliminary step, we label the four latent classes to help clarify the apparent meaning of the classes, we have around 44% of the silver customers i.e. not frequent drinkers of the wine.

The Gold customers are the customers are mid-level drinkers of the wine which are around 17 % in the model. The most important customers are present in class 3. Who could be given some promotions and packages are the Platinum customers and those are around 15%. In last, class 4 represents the Nondrinkers of the wine which are 22%.

Class 1 = Silver customers

Class 2 = Gold Customers

Class 3= Platinum Customers

Class 4 = Non Drinkers

1 2 3 4

0.4450 0.1781 0.1542 0.2227

(0.0317) (0.0244) (0.0230) (0.0265)

**Item Response Plot**

The following code re-runs the 5 class LCA model (with the same seed) and saves an output data set called outseeds\_5c\_alc which serves as input to the %Itemresponseplot macro:

**proc** **lca** data=DATA\_SET\_LC\_RENAMED orig\_weights OUTPARAM = test4 outest=lca1\_outests14 outpost = lca1\_post4;

id id ;

nstarts **300** ;

title " WINE Use: 4 Classes LCA" ;

nclass **4**;

cores **4** ;

items Item1 Item2 Item3 Item4 Item5 Item6 Item7 Item8 Item9 Item10 Item11 Item12;

categories **2** **2** **2** **2** **2** **2** **2** **2** **2** **2** **2** **2**;

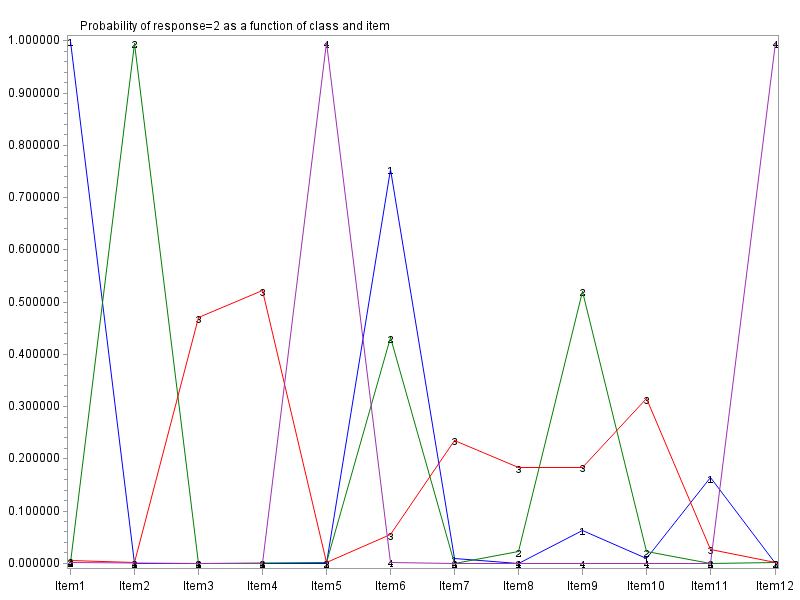
seed **1232** ;

rho prior=**1**;

**run** ;

%INCLUDE 'G:\MS Data Science\3rd semester\Questioner Data Analysis\LcaGraphicsV2-vbywrj\LcaGraphicsV2.sas';

%***ItemResponsePlot***(paramdataset=test4);



The figure above displays item response probabilities for response category 2 on the Y-axis with alcohol behavior variables along the X-axis.

**Model Identification**

The next two figures illustrate use of evaluation tools related to model identification.

Invocation of the model identification macro with the outseeds\_5c\_alc data set produces figure.

/\*Model indentification\*/

%***IdentificationPlot***(seedsdataset=test4\_outseeds) ;



**Size of Classes:**

Once we have come up with a descriptive label for each of the classes, we can look at the number of people who are categorized into each of the classes. I predict that about 44.53% are Silver customer, 17.81% are Gold Customers, and 15% of people are Platinum Customers. I can compare my predictions to the results that SAS produces.

How many gold customers are there? How many Non Drinkers are there? How many Platinum customers are there? One simple way we could determine this is by taking the information from the Class Membership above and doing a simple tabulation on the last column (BEST). Since PROC LCA doesn’t give this to you by default, you can run a simple frequency table using the code below.

|  | **BEST** | | | | |
| --- | --- | --- | --- | --- | --- |
| **Class Name** | **BEST** | **Frequency** | **Percent** | **Cumulative Frequency** | **Cumulative Percent** |
| Silver customers | **1** | 110 | 44.53 | 110 | 44.53 |
| Gold Customers | **2** | 44 | 17.81 | 154 | 62.35 |
| Platinum Customers | **3** | 38 | 15.38 | 192 | 77.73 |
| Non Drinkers | **4** | 55 | 22.27 | 247 | 100.00 |

REFERENCES

Bray, B. C., Lanza. S. T., & Tan, X. (2015). Eliminating bias in classify-analyze approaches for latent class analysis.Structural Equation Modeling: A Multidisciplinary Journal, 22(1), 1-11.

doi: 10.1080/10705511.2014.935265 PMCID: PMC4299667

Collins, L. M., & Lanza, S. T. (2010). Latent class and latent transition analysis: With applications in the social, behavioral, and health sciences. New York: Wiley.

Gorsuch, R. L. (1974), Factor Analysis, Philadelphia: W. B. Saunders.

Lanza, S. T., Bray, B. C., & Collins, L. M. (2013). An introduction to latent class and latent transition analysis.

In J. A. Schinka, W. F. Velicer, & I. B. Weiner (Eds.),Handbook of psychology (2nd ed.,Vol. 2, pp. 691-716). Hoboken, NJ: Wiley