

# Análisis de Clases Latentes

## R + Shiny + poLCA

Investigador

17 de octubre de 2025

### Resumen

Este documento presenta una aplicación Shiny desarrollada en R para realizar Análisis de Clases Latentes (LCA) utilizando el paquete poLCA. La herramienta permite identificar subgrupos homogéneos dentro de poblaciones heterogéneas mediante técnicas de clasificación probabilística aplicadas a variables categóricas.

## 1. Enlace a la Aplicación Shiny

### 1.1. Acceso a la Aplicación Interactiva

<https://milfer.shinyapps.io/EJEMPL01/>

Click en el enlace para acceder a la aplicación interactiva

## 2. Introducción

El Análisis de Clases Latentes (LCA) es una técnica estadística avanzada que identifica subgrupos homogéneos (clases latentes) dentro de una población heterogénea [attached *file* : 1]. *Esta aplicación Shiny proporciona una interfaz completa para realizar análisis LCA automatizados con*

- Clasificación probabilística de observaciones
- Manejo de variables categóricas
- Identificación de perfiles de respuesta
- Comparación de modelos con diferente número de clases
- Visualizaciones interactivas y exportación de resultados

## 3. Fundamento Matemático del LCA

### 3.1. Modelo de Clases Latentes

El modelo básico de LCA asume que la población se compone de  $K$  clases latentes distintas. La probabilidad conjunta de observar un patrón de respuesta específico se expresa como [attached *file* : 1] :

$$P(Y_1, Y_2, \dots, Y_J) = \sum_{k=1}^K \pi_k \prod_{j=1}^J P(Y_j|C_k)$$

donde:

- $\pi_k$  es la proporción de la población en la clase  $k$
- $P(Y_j|C_k)$  es la probabilidad condicional de respuesta en la variable  $j$  dada la clase  $k$
- $J$  es el número de variables observadas

### 3.2. Probabilidades Condicionales

Las probabilidades condicionales de respuesta representan la probabilidad de elegir una categoría específica en una variable dada la pertenencia a una clase [attached<sub>file</sub> : 1] :

$$\rho_{jrk} = P(Y_j = r|C = k)$$

donde  $r$  representa la categoría de respuesta de la variable  $j$ .

## 4. Flujo de Trabajo Recomendado

La aplicación sigue un flujo de trabajo estructurado en seis pasos [attached<sub>file</sub> : 1] :

1. **Cargar Datos:** Importar archivo CSV o utilizar datos de ejemplo
2. **Configurar:** Seleccionar variables y parámetros del modelo
3. **Ejecutar:** Estimar modelo LCA con número de clases especificado
4. **Analizar:** Interpretar resultados y gráficos generados
5. **Comparar:** Evaluar modelos con diferente número de clases
6. **Exportar:** Descargar resultados, tablas y gráficos

## 5. Código de la Aplicación Shiny

### 5.1. Librerías Requeridas

```

1 library(shiny)
2 library(shinydashboard)
3 library(poLCA)
4 library(ggplot2)
5 library(plotly)
6 library(dplyr)
7 library(tidyr)
8 library(DT)
9 library(writexl)
```

Listing 1: Paquetes necesarios para la aplicación

## 5.2. Interfaz de Usuario (UI)

La interfaz está estructurada utilizando `shinydashboard` con múltiples pestañas para diferentes funcionalidades [attached *file* : 1] :

```
1 ui <- dashboardPage(  
2   dashboardHeader(title = "An lisis de Clases Latentes"),  
3  
4   dashboardSidebar(  
5     sidebarMenu(  
6       menuItem("Inicio", tabName = "inicio", icon = icon("home"))  
7       ,  
8       menuItem("Cargar Datos", tabName = "datos", icon = icon("upload")),  
9       menuItem("An lisis", tabName = "analisis", icon = icon("chart-bar")),  
10      menuItem("Resultados", tabName = "resultados", icon = icon("table")),  
11      menuItem("Comparaci n", tabName = "comparacion", icon = icon("balance-scale")),  
12      menuItem("Exportar", tabName = "exportar", icon = icon("download")),  
13      menuItem("Gu a", tabName = "guia", icon = icon("info-circle"))  
14    )  
15  ),  
16  dashboardBody(  
17    # Contenido de cada pesta a  
18    tabItems(  
19      # Pesta a de inicio  
20      tabItem(tabName = "inicio",  
21        h2(" Qu es el An lisis de Clases Latentes?"),  
22        # Contenido informativo  
23      ),  
24      # Otras pesta as...  
25    )  
26  )  
27 )
```

Listing 2: Estructura básica de la UI

## 5.3. Servidor (Server)

El servidor maneja la lógica de análisis LCA [attached *file* : 1][*file* : 2] :

```
1 server <- function(input, output, session) {  
2  
3   # Reactive values para almacenar datos y resultados  
4   datos_reactive <- reactiveVal(NULL)  
5   modelo_lca <- reactiveVal(NULL)  
6  
7   # Cargar datos
```

```

8  observeEvent(input$file, {
9    req(input$file)
10   datos <- read.csv(input$file$datapath)
11   datos_reactive(datos)
12 })
13
14 # Ejecutar análisis LCA
15 observeEvent(input$ejecutar_lca, {
16   req(datos_reactive())
17
18   # Preparar fórmula
19   variables <- input$variables_seleccionadas
20   formula <- as.formula(paste("cbind(",
21                               paste(variables, collapse = ","),
22                               ") ~ 1"))
23
24   # Estimar modelo
25   lca_model <- poLCA(formula = formula,
26                      data = datos_reactive(),
27                      nclass = input$num_clases,
28                      nrep = 10,
29                      maxiter = 3000)
30
31   modelo_lca(lca_model)
32 })
33
34 # Generar salidas y visualizaciones
35 # ...
36 }

```

Listing 3: Estructura del servidor

## 6. Índices de Bondad de Ajuste

El modelo LCA utiliza múltiples criterios para evaluar la calidad del ajuste [attached *file* : 1] :

### 6.1. AIC (Criterio de Información de Akaike)

$$\text{AIC} = -2 \log L + 2p$$

donde  $\log L$  es el log-likelihood y  $p$  es el número de parámetros estimados [attached *file* : 1].

### 6.2. BIC (Criterio de Información Bayesiano)

$$\text{BIC} = -2 \log L + p \log(n)$$

donde  $n$  es el tamaño de la muestra. Valores menores de AIC y BIC indican mejor ajuste [attached<sub>file</sub> : 1].

### 6.3. Log-Likelihood

$$\log L = \sum_{i=1}^n \log P(Y_i|\theta)$$

Valores más altos (menos negativos) indican mejor ajuste del modelo [attached<sub>file</sub> : 1].

## 7. Aplicaciones del LCA

La técnica tiene múltiples aplicaciones en diferentes campos [attached<sub>file</sub> : 1] :

Campo	Aplicación
Salud	Identificación de perfiles de pacientes y patrones de síntomas
Educación	Clasificación de estilos de aprendizaje y rendimiento académico
Marketing	Segmentación de consumidores y perfiles de comportamiento
Psicología	Tipologías de personalidad y patrones conductuales

Cuadro 1: Aplicaciones del Análisis de Clases Latentes

## 8. Requisitos de Datos

### 8.1. Datos Válidos

Para un análisis exitoso, los datos deben cumplir las siguientes condiciones [attached<sub>file</sub> : 1] :

- Variables categóricas (no continuas)
- Valores comenzando en 1 (no en 0)
- Sin valores faltantes (NA)
- Formato CSV con encabezados

### 8.2. Recomendaciones

- Mínimo 200 observaciones
- 3-10 variables idealmente
- Categorías balanceadas
- Preprocesar datos en R antes de cargar

## 9. Interpretación de Resultados

### 9.1. Probabilidades Condicionales

Valores mayores a 0.7 indican fuerte asociación de la clase con esa respuesta específica [attached<sub>file</sub> : 1]. *Perfiles coherentes facilitan la interpretación de las clases latentes identificadas* [attached<sub>file</sub> : 1].

### 9.2. Entropía

La entropía mide la incertidumbre en la clasificación [attached<sub>file</sub> : 1] :

$$E = 1 - \frac{\sum_{i=1}^n \max_k P(C_k|Y_i)}{n}$$

Valores cercanos a 0 indican clasificación clara y precisa [attached<sub>file</sub> : 1].

## 10. Visualizaciones Implementadas

La aplicación genera múltiples visualizaciones interactivas [attached<sub>file</sub> : 1] :

- Distribución de clases latentes (gráfico de barras)
- Probabilidades condicionales de respuesta por variable
- Matriz de probabilidades (heatmap)
- Probabilidades posteriores de clasificación
- Análisis de entropía por clase
- Gráficos comparativos de índices AIC y BIC
- Evolución del log-likelihood

## 11. Exportación de Resultados

La aplicación permite exportar [attached<sub>file</sub> : 1] :

### 11.1. Conjuntos de Datos

- Datos originales (CSV)
- Datos con clasificación asignada (CSV y Excel)

### 11.2. Reportes y Análisis

- Probabilidades condicionales (CSV)
- Comparación de modelos (CSV)
- Índices de ajuste (CSV)

### 11.3. Gráficos

- Probabilidades condicionales (PNG)
- Comparación de modelos (PNG)
- Gráficos interactivos exportables

## 12. Conclusión

Esta aplicación Shiny proporciona una herramienta completa y accesible para realizar Análisis de Clases Latentes [attached<sub>file</sub> : 1]. Las características principales incluyen [attached<sub>file</sub> : 1] :

1. Interfaz intuitiva y fácil de usar
2. Análisis completo automatizado
3. Múltiples visualizaciones interactivas
4. Comparación sistemática de modelos
5. Exportación completa de resultados
6. Documentación integrada y guías de uso

El uso de poLCA en combinación con Shiny permite democratizar el acceso a técnicas estadísticas avanzadas, facilitando la identificación de perfiles latentes en diversas áreas de investigación [attached<sub>file</sub> : 1].

## 13. Enlace Directo a la Aplicación

<https://milfer.shinyapps.io/EJEMPL01/>

Click para acceder a la herramienta interactiva de Análisis de Clases Latentes