Methods and Applications of Finite Mixture Models, with Computing Demonstrations Using the R Package `mixtools`

Kedai Cheng (University of North Carolina - Asheville)
Derek S. Young (University of Kentucky)

Conference on Statistical Practice (CSP) 2022
February 3rd, 2022

College of Arts and Sciences
Dr. Bing Zhang Department of Statistics

University of North Carolina Asheville

Funding Acknowledgment

This work is supported by the Chan Zuckerberg Initiative: Essential Open Source Software for Science (Grant Number 2020-225193).

Before We Begin...

- You can download this presentation file and the corresponding R scripts from my GitHub repo: https://github.com/dsy109/Supplemental/tree/main/CSP 2022 Trainings/mixtools
- If you plan to run the R examples concurrently, make sure you have the most recent versions installed for all of the packages listed at the top of the .R file
- This tutorial was developed and tested using R version 4.1.1

About the Instructor

- PhD in Statistics from Penn State (2007)
  - Research on finite mixtures-of-regressions models
- Worked for Naval Nuclear Propulsion Program (NNPP) at Bettis Atomic Power Laboratory in Pittsburgh (2008-2011)
  - Researched and applied novel statistical quality control methods and tolerance regions
- Worked for U.S. Census Bureau in Washington, DC (2011-2014)
  - Survey data analysis with tolerance regions and zero-inflated models
- Taught online for Penn State (2008-2013)
- UK Department of Statistics (2014-Present)

Purpose of Tutorial

- Applications that use finite mixture models are found in nearly every field
- Of course, the availability of computational tools and resources are crucial to doing analysis with mixture models
- The `R` package `mixtools` (Benaglia et al., 2009) is a leading software package in this respect.
- This highly-cited package (1200+ citations according to Google Scholar) has been used to analyze diverse research questions involving quasars data, maize production, clustering of patients with leukemia, subpopulations of individuals with autism and schizophrenia, and certain color variants on manta rays
- The major goals of this PCD will be to (1) inform the attendees as to best practices when using finite mixture models and (2) gain proficiency with tools available in the `mixtools` package

Outline of Topics

Preliminaries

`mixtools`

- Gaussian Mixture Models
- Parametric and Semiparametric Mixtures of Regressions
- Other Parametric Mixture Models
- Determining the Number of Components
- Visualizing Estimated Mixture Models

Final Comments

Preliminaries

Motivation: Finite Mixture Models

- **Finite mixture models** are used to model data where the observations are sampled from a population that consists of several homogeneous subpopulations (called the **components** of the population), but to which subpopulation each observation belongs is unknown
- Estimation of mixture components is an unsupervised learning task
- Make a *soft* probabilistic classification of each observation to a component, whereas cluster analysis performs a *hard* classification to a cluster
  - Finite mixture models are, thus, naturally used as the underlying models for model-based clustering
- Can be used for density estimation and viewed as a kind of kernel method
- Used for numerous interesting problems in areas like pattern recognition, finance, psychology, and astronomy

Finite Mixture Models

Definition

The random vector $\mathbf{Y} \in \mathbb{R}^p$ follows a parametric **mixture distribution** if it has the mixture density

$$f(\mathbf{y}|\boldsymbol{\psi}) = \sum_{j=1}^{k} \lambda_j g_j(\mathbf{y}|\boldsymbol{\theta}_j), \tag{1}$$

where $k \in \mathbb{N}$ is the number of components, which ideally should be known *a priori*, the λ_js are **mixing proportions** that comprise the standard simplex

$$\left\{ \boldsymbol{\lambda} \in \mathbb{R}^k : \sum_{j=1}^{k} \lambda_j = 1, \lambda_j > 0, j = 1, \ldots, k \right\},$$

and the g_j are component-specific densities from a parametric family with parameter $\boldsymbol{\theta}_j \in \Theta_j \subseteq \mathbb{R}^q$, such that Θ_j is open in $\mathbb{R}^q$. Usually, $g_j \equiv g$. The mixture density f in (1) is, thus, parameterized by $\boldsymbol{\psi} = (\lambda_1, \ldots, \lambda_{k-1}, \boldsymbol{\theta}_1^\top, \ldots, \boldsymbol{\theta}_k^\top)^\top \in \boldsymbol{\Psi} \subset \mathbb{R}^{k(q+1)-1}$.

- When $\mathbf{X}$ is a set of covariates for a univariate response Y, then the preceding framework is easily extended to model $Y|\mathbf{X}$ using a mixture structure, giving us what are called **mixtures-of-regressions models**

Overview of the EM Algorithm

- Maximum likelihood is the dominant form of estimation in applied statistics
- Because closed-form solutions to likelihood equations are the exception rather than the rule, we need effective numerical recipes to perform estimation
- EM algorithms (where the EM stands for **expectation-maximization**) are iterative methods to find maximum likelihood estimators of models that have missing data or depend on some sort of unobserved latent variables which are treated as missing data
- The EM algorithm is one of the most popular algorithms in statistics, as noted by the citation count for the original paper by Dempster, Laird, and Rubin (1977), which is over $65,000$

The EM as a Family of Algorithms

- EM algorithms, at a high level, have two steps: the **E-step** (**expectation-step**) and the **M-step** (**maximization-step**)
- The EM algorithm is not so much an algorithm as a methodology for creating a family of algorithms
- There are a number of canonical problems where now an EM-type algorithm is the standard approach, including in the estimation of mixture models
- The basic idea underlying the EM algorithm is as follows:
  - We observe some data that we represent with $\mathbf{Y}$
  - However, there are some missing data, that we represent with $\mathbf{Z}$
  - Together, the observed data $\mathbf{Y}$ and the missing data $\mathbf{Z}$ make up the complete data $\mathbf{X} = (\mathbf{Y}, \mathbf{Z})$
    - Note that the above setup is not being rigorous with respect to dimensions of the quantities as this is intended to just give a general setup for problems where EM can be applied

Setup for Using EM Algorithms

- We assume that the complete data have a density $g(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta})$ that is parameterized by the vector of parameters $\boldsymbol{\theta}$. Because of the missing data, we cannot evaluate g
- The observed data have the density

$$f(\mathbf{y}|\boldsymbol{\theta}) = \int g(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta}) d\mathbf{z}$$

  and the **observed data loglikelihood** is $\ell_o(\boldsymbol{\theta}; \mathbf{y}) = \log f(\mathbf{y}|\boldsymbol{\theta})$
- The problem now is that $\ell_o(\boldsymbol{\theta}; \mathbf{y})$ is difficult to evaluate or maximize because of the integral (or a sum for discrete problems); however, in order to estimate $\boldsymbol{\theta}$ via maximum likelihood using only the observed data, we need to be able to maximize $\ell_o(\boldsymbol{\theta}; \mathbf{y})$
- The complete data density usually has some nice form (like being an exponential family member) so that if we had the missing data $\mathbf{Z}$, we could easily evaluate $g(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta})$. Then, the **complete data loglikelihood** is $\ell_c(\boldsymbol{\theta}; \mathbf{y}, \mathbf{z}) = \log g(\mathbf{y}, \mathbf{z}|\boldsymbol{\theta})$

The EM Algorithm

1. *E-step*: Let $\boldsymbol{\theta}^{(t)}$ be the current estimate of $\boldsymbol{\theta}$. Calculate the following:

$$Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}) \triangleq \mathrm{E}_{\mathbf{Z}}[\ell_c(\boldsymbol{\theta}; \mathbf{Y}, \mathbf{Z}) | \mathbf{Y}, \boldsymbol{\theta}^{(t)}].$$

2. *M-step*: Find the parameter that maximizes the following:

$$\boldsymbol{\theta}^{(t+1)} = \arg\max_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} Q(\boldsymbol{\theta}; \boldsymbol{\theta}^{(t)}).$$

3. Iterate between the E-step and M-step until a convergence criterion is achieved.

EM Algorithms for Finite Mixture Models

- To construct an EM algorithm for mixtures, we first introduce the (unobservable) indicator random variable

$$\mathbf{Z}_{i,j} = \mathtt{I}\{\text{observation } i \text{ belongs to component } j\},$$

  for $i = 1, \ldots, n$ and $j = 1, \ldots, k$

- The measurements $\mathbf{Y}$ in our previous mixture model definition are the observed (incomplete) data and $\mathbf{X} = (\mathbf{Y}, \mathbf{Z})$ are, again, the complete data
- The observed and complete data loglikelihoods are, respectively,

$$\ell_o(\boldsymbol{\psi}) = \log L(\boldsymbol{\psi}; \mathbf{y}) = \sum_{i=1}^{n} \log f(\mathbf{y}_i|\boldsymbol{\psi}) \quad \text{and}$$

$$\ell_c(\boldsymbol{\psi}) = \log \prod_{i=1}^{n} \prod_{j=1}^{k} [\lambda_j g(\mathbf{y}_i|\boldsymbol{\theta}_j)]^{\mathbf{Z}_{i,j}} = \sum_{i=1}^{n} \sum_{j=1}^{k} \mathbf{Z}_{i,j} \log[\lambda_j g(\mathbf{y}_i|\boldsymbol{\theta}_j)]$$

EM Algorithms for Finite Mixture Models (ctd.)

- Notice $\mathbf{Z}_{i,j} \sim \textit{Bern}(\lambda_j)$ and $\mathbf{Z}_{i,j}$ is independent of $\mathbf{Y}_{i^*}$ for all $i^* \neq i$
- Since $\mathsf{E}_{\boldsymbol{\psi}^{(t)}}$ is a linear functional, the right-hand side of the conditional expectation in the E-step, combined with the separability of the components in $\ell_c(\boldsymbol{\psi})$ allows us to replace $\mathbf{Z}_{i,j}$ by

$$\mathsf{E}_{\boldsymbol{\psi}}[\mathbf{Z}_{i,j}|\mathbf{Y} = \mathbf{y}] = \frac{\lambda_j g(\mathbf{y}_i|\boldsymbol{\theta}_j)}{\sum_{l=1}^{k} \lambda_l g(\mathbf{y}_i|\boldsymbol{\theta}_l)},$$

  which follows from an application of Bayes' rule and the law of total probability
- Thus, when provided the estimate $\boldsymbol{\psi}^{(t)}$, we get

$$\mathbf{z}_{i,j}^{(t)} = \frac{\lambda_j^{(t)} g(\mathbf{y}_i|\boldsymbol{\theta}_j^{(t)})}{\sum_{l=1}^{k} \lambda_l^{(t)} g(\mathbf{y}_i|\boldsymbol{\theta}_l^{(t)})},$$

  which we call **posterior membership probabilities**
- Also note in the E-Step that the expectation of the complete data log likelihood is conditioned on the observed data and that it does not strictly replace missing data by their conditional expectations

Estimated Standard Errors

- With likelihood methods, it is possible to obtain standard error estimates by using the inverse of the observed information matrix when implementing a Newton-type method; however, this is often computationally burdensome in estimation of mixture models
- In `mixtools`, we offer the `boot.se()` function to compute standard errors in the likelihood setting by implementing a parametric bootstrap
- The parametric bootstrap should provide similar estimates to the standard errors compared to the method involving the information matrix
- We outline the algorithm for a parametric bootstrapping scheme in the mixture setting using an EM algorithm on the next slide

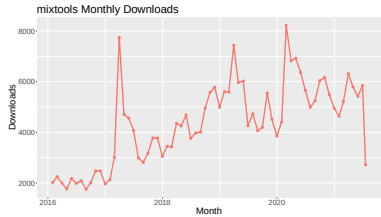
Bootstrapping for Estimating Standard Errors

Parametric Bootstrap for Estimating Standard Errors

1. Find the maximum likelihood estimate $\hat{\psi}$ by implementing an EM algorithm based on the values $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$.
2. Generate a bootstrap sample of size n from $f(\mathbf{y}; \hat{\psi})$ and call this sample $\mathbf{y}_1^*, \mathbf{y}_2^*, \ldots, \mathbf{y}_n^*$.
3. Find the estimate $\tilde{\psi}$ for the bootstrap sample by implementing an EM algorithm.
4. Repeat steps 2 and 3 B times to generate the bootstrap sampling distribution $\tilde{\psi}^{(1)}, \tilde{\psi}^{(2)}, \ldots, \tilde{\psi}^{(B)}$.
5. Compute the bootstrap variance-covariance matrix as the sample variance-covariance matrix of the generated values $\tilde{\psi}^{(1)}, \tilde{\psi}^{(2)}, \ldots, \tilde{\psi}^{(B)}$.


mixtools

- First release: 2006
- The package includes: functions for estimating classic mixture models (e.g., Gaussian mixture models and mixtures-of-regressions models), non/semiparametric procedures for mixture analysis, visualization tools, bootstrapping routines for estimating standard errors, and model selection routines
- The figure below shows a very strong increasing trend over time; the average number of monthly downloads per year is approximately 2080, 3819, 4436, 5151, and 5942, in 2016, 2017, 2018, 2019, and 2020, respectively



General Capabilities of `mixtools`

Parametric Procedures

- Classic parametric distributions
  - (Multivariate) normal [`normalmixEM`, `mvnormalmixEM`]
  - Gamma [`gammamixEM`]
  - Multinomial [`multmixEM`]
- Parametric regression settings
  - (Piecewise) linear regression [`regmixEM`, `regmixMH`, `segregmixEM`]
  - Logistic regression [`logisregmixEM`]
  - Poisson regression [`poisregmixEM`]

Nonparametric Procedures

- Individual repeated measures [`npEM`, `npMSL`]
- Conditionally independent multivariate repeated measures [`mvnpEM`]

Semiparametric Procedures

- Univariate symmetric location densities [`spEMsymloc`, `spEMsymlocN01`]
- Linear regressions with unspecified error structure [`spregmix`]
- Scaled mixture of censored data [`spRMM_SEM`]

General Capabilities of `mixtools` (ctd.)

Secondary Procedures

- Bootstrapping for standard errors [`boot.se`]
- Bootstrapping to determine number of components [`boot.comp`]
- Model selection routines [`regmixmodel.sel`, `multmixmodel.sel`]

Visualizations

- Mixture density estimates [`plot.mixEM`, `plot.mvnpEM`]
- Mixturegram [`mixturegram`]
- FDR estimates from EM-like strategies [`plotFDR`]

Format of Typical `mixtools` Function

```
foo_EM(data, k, theta, conv.crit)
```

Function Inputs

- `data`: Often a vector, matrix, or data frame
- `k`: The number of components to fit for the mixture; an integer greater than 1
- `theta`: Optional starting values for the parameters in the model, otherwise a starting value strategy is employed
- `conv.crit`: Controls for determining convergence of the EM algorithm

Function Outputs

- Final parameter estimates; i.e., the maximum likelihood estimates
- Loglikelihood values to monitor convergence or use in model selection criteria calculations (e.g., AIC, BIC)
- Posterior membership probabilities of each observation (subject); can be used for model-based clustering

Gaussian Mixture Models

Gaussian Mixture Models

- Perhaps the most commonly-used class of mixture models
- The observed univariate data $y_1, \ldots, y_n$ are sampled from a k-component Gaussian mixture if they have the density

$$f(y|\boldsymbol{\psi}) = \sum_{j=1}^{k} \lambda_j \phi(y; \mu_j, \sigma_j^2),$$

where $\phi(\cdot; \mu, \sigma^2)$ is the normal density with mean μ and variance σ^2 and $\boldsymbol{\psi} = (\lambda_1, \ldots, \lambda_{k-1}, \mu_1, \ldots, \mu_k, \sigma_1^2, \ldots, \sigma_k^2)^{\mathrm{T}}$

- The observed p-dimensional multivariate data $\mathbf{y}_1, \ldots, \mathbf{y}_n$ are sampled from a k-component multivariate Gaussian mixture if they have the density

$$f(\mathbf{y}|\boldsymbol{\theta}) = \sum_{j=1}^{k} \lambda_j \phi_p(y; \boldsymbol{\mu}_j, \Sigma_j),$$

where $\phi_p(\cdot; \boldsymbol{\mu}_j, \Sigma_j)$ is the p-dimensional multivariate normal density with mean vector $\boldsymbol{\mu}$ and variance-covariance matrix Σ and $\boldsymbol{\theta} = (\lambda_1, \ldots, \lambda_{k-1}, \boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_k, \Sigma_1, \ldots, \Sigma_k)^{\mathrm{T}}$

2-Component Gaussian Mixture

- The observed data $y_1, \ldots, y_n$ are sampled from a 2-component Gaussian mixture if they have the density

$$f(y|\boldsymbol{\psi}) = \lambda\phi(y; \mu_1, \sigma_1^2) + (1-\lambda)\phi(y; \mu_2, \sigma_2^2),$$

where $\boldsymbol{\psi} = (\lambda, \mu_1, \mu_2, \sigma_1^2, \sigma_2^2)^{\mathrm{T}}$

- The observed data loglikelihood is

$$\ell_o(\boldsymbol{\psi}) = \sum_{i=1}^{n} \log\left\{\lambda\phi(y_i; \mu_1, \sigma_1^2) + (1-\lambda)\phi(y_i; \mu_2, \sigma_2^2)\right\}$$

- Letting $Z_i \sim Bern(\lambda)$, the complete data loglikelihood is

$$\ell_c(\boldsymbol{\psi}) = \sum_{i=1}^{n}\Big\{ Z_i \log\phi(y_i; \mu_1, \sigma_1^2) + (1-Z_i)\log\phi(y_i; \mu_2, \sigma_2^2) + Z_i\log\lambda + (1-Z_i)\log(1-\lambda)\Big\}$$

2-Component Gaussian Mixture (ctd.)

- For the E-step at iteration t, $t = 0, 1, \ldots$, calculate

$$z_i^{(t)} = \frac{\lambda^{(t)} \phi(y_i | \mu_1^{(t)}, \sigma_1^{2(t)})}{\lambda^{(t)} \phi(y_i | \mu_1^{(t)}, \sigma_1^{2(t)}) + (1 - \lambda^{(t)}) \phi(y_i | \mu_2^{(t)}, \sigma_2^{2(t)})}$$

- The formulas for maximizing our objective function in the M-step are

$$\mu_1^{(t+1)} = \frac{\sum_{i=1}^n z_i^{(t)} y_i}{\sum_{i=1}^n z_i^{(t)}}, \qquad \mu_2^{(t+1)} = \frac{\sum_{i=1}^n (1 - z_i^{(t)}) y_i}{\sum_{i=1}^n (1 - z_i^{(t)})},$$

$$\sigma_1^{2(t+1)} = \frac{\sum_{i=1}^n z_i^{(t)} (y_i - \mu_1^{(t+1)})^2}{\sum_{i=1}^n z_i^{(t)}}, \qquad \sigma_2^{2(t+1)} = \frac{\sum_{i=1}^n (1 - z_i^{(t)}) (y_i - \mu_2^{(t+1)})^2}{\sum_{i=1}^n (1 - z_i^{(t)})},$$

$$\lambda^{(t+1)} = n^{-1} \sum_{i=1}^n z_i^{(t)}$$

Example: Quasars Data

- Quasars are extremely luminous objects in the Universe that arise from the accretion of gas onto supermassive black holes in the center of a galaxy
- The analysis of absorption lines in quasar spectra aid in the study of metal-enriched environments
- We consider the normalized intensity of the quasar light for the 3-times-ionized silicon line Si IV 1394 for the $z = 0.653411$ absorption system
- This dataset consists of $n = 104$ measurements
- We will model these data with a 2-component Gaussian mixture, which was the final model selected in Young, Ke, and Zeng (2018)
- Source: J. C. Charlton et al. (2013), "High-Resolution STIS/Hubble Space Telescope and HIRES/Keck Spectra of Three Weak Mg II Absorbers Toward PG 1634 + 706." *The Astrophysical Journal*; 589(1):111–125.

Example: Hidalgo Stamp Data

- The Hidalgo stamp dataset contains $n = 485$ records of the thickness of stamps having images of Miguel Hidalgo y Costilla, a famous leader of the Mexican War of Independence
- The stamps were issued by Mexico in 1872 and circulated until 1874
- Due to poor quality control at that time, the thicknesses of the stamps varied considerably
- These data have been extensively analyzed using both nonparametric approaches and Gaussian mixtures in order to identify different components
- Source: A. J. Izenman and C. J. Sommer (1988), "Philatelic Mixtures and Multimodal Distributions." *Journal of the American Statistical Association*; 83(404):941–953.

Example: Diffuse Large B-Cell Lymphoma

- Three biomarkers (CD3, CD5, and CD19) were measured on cells derived from the lymph nodes of patients diagnosed with Diffuse Large B-Cell Lymphoma (DLBCL)
- This dataset is for one subject with 8183 biomarkers
- Analyzed a subset of $n = 500$ observations to identify possible clusters of biomarkers
- We will model using a 2-component bivariate Gaussian mixture and show various visualizations
- Source: N. Aghaeepour et al. (2013), "Critical Assessment of Automated Flow Cytometry Data Analysis Techniques." *Nature Methods*; 10(3):228–238.

Parametric and Semiparametric Mixtures of Regressions

Mixtures of Linear Regressions

- The observed data $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$ follow a k-component mixture-of-linear-regressions model if they have the density

$$f(y|\mathbf{x}, \boldsymbol{\psi}) = \sum_{j=1}^{k} \lambda \phi(y; \mathbf{x}^{\mathrm{T}}\boldsymbol{\beta}_j, \sigma_j^2),$$

  which is just the Gaussian mixture density with the mean μ_j replaced by the simple linear regression relationship $\mathbf{x}^{\mathrm{T}}\boldsymbol{\beta}_j$, $j = 1, \ldots, k$
- The observed data loglikelihood is

$$\ell_o(\boldsymbol{\psi}) = \sum_{i=1}^{n} \log \left\{ \sum_{j=1}^{k} \lambda_j \phi(y_i; \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}_j, \sigma_j^2) \right\}$$

- Letting $Z_{ij} \sim Bern(\lambda_j)$, the complete data loglikelihood is

$$\ell_c(\boldsymbol{\psi}) = \sum_{i=1}^{n} \sum_{j=1}^{k} Z_{ij} \log \phi(y_i; \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}_j, \sigma_j^2)$$

Mixtures of Linear Regressions (ctd.)

- For the E-step at iteration t, $t = 0, 1, \ldots$, calculate

$$z_{ij}^{(t)} = \frac{\lambda_j^{(t)} \phi(y_i | \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}_j, \sigma_j^{2(t)})}{\sum_{l=1}^{k} \lambda_l^{(t)} \phi(y_i | \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}_l, \sigma_l^{2(t)})}$$

- Letting $\mathbf{W}_j^{(t)} = \mathrm{diag}(z_{1j}^{(t)}, \ldots, z_{nj}^{(t)})$, $\|\mathbf{A}\|^2 = \mathbf{A}^{\mathrm{T}}\mathbf{A}$, and using vector/matrix notation for all relevant quantities, the formulas for maximizing our objective function in the M-step are

$$\begin{aligned}
\boldsymbol{\beta}_j^{(t+1)} &= (\mathbf{X}^{\mathrm{T}} \mathbf{W}_j^{(t)} \mathbf{X})^{-1} \mathbf{X}^{\mathrm{T}} \mathbf{W}_j^{(t)} \mathbf{y} \\
\sigma_j^{2(t+1)} &= \frac{\left\| \mathbf{W}_j^{1/2(t)} (\mathbf{y} - \mathbf{X}^{\mathrm{T}} \boldsymbol{\beta}_j^{(t+1)}) \right\|^2}{\mathrm{tr}(\mathbf{W}_j^{(t)})} \\
\lambda_j^{(t+1)} &= n^{-1} \mathrm{tr}(\mathbf{W}_j^{(t)})
\end{aligned}$$

Example: Carbon Dioxide Data

- For several years, data have been collected on the carbon dioxide emissions of most sovereign states and territories
- We will analyze a dataset of the gross national product (GNP) per capita in 1996 for $n = 28$ countries as well as their estimated carbon dioxide (CO_2) emission per capita for the same year
- As we will see, the data demonstrate possibly two different regression relationships that could be present; thus, we will fit a 2-component mixture-of-linear-regressions model, which was also explored in Hurn, Justel, and Robert (2003)
- Source: C. M. Hurvich, J. S. Simonoff, and C.-L. Tsai (1998), "Smoothing Parameter Selection in Nonparametric Regression Using an Improved Akaike Information Criterion." *Journal of the Royal Statistical Society, Series B*; 60(2):271–293.

Semiparametric Mixtures of Regressions

- Now consider the model

$$g(y_i|\mathbf{x}_i) = \sum_{j=1}^{k} \lambda_j f(y_i - \mathbf{x}_i^{\mathrm{T}}\boldsymbol{\beta}_j),$$

  where f is an unspecified density function
  - The only assumption is that the errors are iid
- The parameters of interest are the λ_js, $\boldsymbol{\beta}_j$s, and f
- This model provides more flexibility in attempting to characterize the different regression relationships
- Estimation must be adapted to include an estimate of f
- There are also some interesting issues concerning identifiability of the model parameters, which we will not discuss here; see Hunter and Young (2012)

"EM-like" Algorithm: E-Step

- Estimation is performed via an "EM-like" algorithm, so named because we retain the E-step and M-step characteristic of a true EM algorithm
  - This algorithm is employed by the `spregmix()` function
- For $t = 0, 1, 2, \ldots$, the E-step consists of finding the posterior membership probabilities:

$$z_{ij}^{(t)} = \frac{\lambda_j^{(t)} f^{(t)}(y_i - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}_j^{(t)})}{\sum_{\ell=1}^{k} \lambda_\ell^{(t)} f^{(t)}(y_i - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}_\ell^{(t)})}$$

"EM-like" Algorithm: M-Step

- The M-step consists of updating the Euclidean parameters
  - The mixing proportions are updated as:

$$\lambda_j^{(t)} = \frac{1}{n} \sum_{i=1}^{n} z_{ij}^{(t)}$$

  - In an ordinary EM algorithm, we update $\boldsymbol{\beta}_j$ by maximizing an expected conditional log-likelihood function; for the present setting, there is no obvious choice, but possibilities include:
    - $\boldsymbol{\beta}_j^{(t)} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} z_{ij}^{(t)} (y_i - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta})^2$
    - $\boldsymbol{\beta}_j^{(t)} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} z_{ij}^{(t)} |y_i - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}|$
    - $\boldsymbol{\beta}_j^{(t)} = \arg\min_{\boldsymbol{\beta}} \sum_{i=1}^{n} z_{ij}^{(t)} f^{(t)} (y_i - \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta})$

"EM-like" Algorithm: Density Estimation Step

- For some bandwidth h and kernel density $K(\cdot)$, update the estimate of f as:

$$f^{(t)}(u) = \frac{1}{nh} \sum_{i=1}^{n} \sum_{j=1}^{k} z_{ij}^{(t)} K\left(\frac{u - y_i + \mathbf{x}_i^{\mathrm{T}} \boldsymbol{\beta}_j^{(t)}}{h} \right)$$

- It is possible to update f by enforcing certain constraints, such as:
  - f must have zero mean or median
  - f must be symmetric about zero
- It is also possible to incorporate a bandwidth update step

Maximized Smoothed Likelihood Estimation

- The "EM-like" algorithm we presented is not a true EM algorithm since there is no likelihood function that may be shown to increase at each iteration
- By applying the work of Levine, Hunter, and Chauveau (2011), our algorithm can be adapted to produce a new algorithm that does increase the value of a smoothed version of the loglikelihood at each iteration
- The nonlinear smoothing operator

$$\mathcal{N}_h f(x) = \exp \int \frac{1}{h} K\left(\frac{x-u}{h}\right) \log f(u) du$$

  can be used to define a smoothed version of the loglikelihood
- Incorporation of the smoothed loglikelihood into our earlier algorithm can then be shown to possess the ascent property

Example: Aphids Data

- A question of interest to plant scientists is how quickly an infection, spread by insects, will propagate through a population of previously healthy plants
- A study was conducted to assess the spread of a viral infection among potato plants by aphids
- $n = 51$ batches of aphids were released over the potato plants (predictor) and the number of infected plants (response) was recorded
- The data exhibit a bifurcation into two linear components, so we will explore 2-component semiparametric mixtures of regressions
- Source: T. R. Turner (2000), "Estimating the Propagation Rate of a Viral Infection of Potato Plants via Mixtures of Regressions." *Journal of the Royal Statistical Society, Series C*; 49(3):371–384.

Other Parametric Mixture Models

Mixtures of Gammas

- Shape-scale parameterization of the gamma distribution has probability density function (pdf)

$$g(x;\alpha,\beta) = \frac{x^{\alpha-1}e^{-x/\beta}}{\beta^{\alpha}\Gamma(\alpha)} \quad \text{for } x > 0,$$

  where $\Gamma(z) = \int_0^\infty t^{z-1}e^{-t}dt$ is the (complete) gamma function, $\alpha > 0$ is a shape parameter, and $\beta > 0$ is a scale parameter
- Allowing each of the k components to have their own shape and scale parameter yields the mixture-of-gammas distribution, with pdf

$$f(x;\boldsymbol{\psi}_1) = \sum_{j=1}^{k} \lambda_j g(x;\alpha_j,\beta_j),$$

  where $\boldsymbol{\psi}_1 = (\lambda_1,\ldots,\lambda_{k-1},\alpha_1,\ldots,\alpha_k,\beta_1,\ldots,\beta_k)^{\mathrm{T}} \in \boldsymbol{\Psi}_1 \subset \mathbb{R}^{3k-1}$
- Assuming a common shape yields

$$f(x;\boldsymbol{\psi}_2) = \sum_{j=1}^{k} \lambda_j g(x;\alpha,\beta_j),$$

  where now $\boldsymbol{\psi}_2 = (\lambda_1,\ldots,\lambda_{k-1},\alpha,\beta_1,\ldots,\beta_k)^{\mathrm{T}} \in \boldsymbol{\Psi}_2 \subset \mathbb{R}^{2k}$; $\alpha = 1$ is a mixtures of exponentials

Example: Whole Genome Duplication Data

- Whole genome duplication (WGD) — defined as the simultaneous gaining of extra copies of all the nuclear chromosomes of an organism — is one of the main processes than can lead to the existence of polyploid organisms
- Synonymous substitutions per synonymous site (Ks) plots, representing the distribution of within-taxon synonymous distances among paralogs genes, have been used as a reliable way to visualize WGD events, with multiple peaks representing multiple successive WGDs
- The Ks plot for the species *Pereskia aculeata (Cactaceae)* was generated using transcriptomic public data; the size of this dataset is $n = 2618$
- Source: Y. Yang et al. (2018), "Improved Transcriptome Sampling Pinpoints 26 Ancient and More Recent Polyploidy Events in Caryophyllales, Including Two Allopolyploidy Events." *New Phytologist*; 217(2):855–870.

Mixtures of Poissons (and Poisson Regressions)

- The Poisson distribution has probability mass function (pmf)

$$g(y;\pi) = \frac{\pi^y \exp\{-\pi\}}{y!} \quad \text{for } y = 0, 1, 2, \ldots,$$

  where $\pi > 0$ is a rate parameter
- Allowing each of the k components to have their own rate parameter yields the mixture-of-Poisson distribution, with pdf

$$f(y;\boldsymbol{\psi}_1) = \sum_{j=1}^{k} \lambda_j g(y;\pi_j),$$

  where $\boldsymbol{\psi}_1 = (\lambda_1, \ldots, \lambda_{k-1}, \pi_1, \ldots, \pi_k)^{\mathrm{T}} \in \boldsymbol{\Psi}_1 \subset \mathbb{R}^{2k-1}$
- If each y is measured with a set of p-dimensional predictors $\mathbf{x}$, and a mixture structure is assumed, then we can relate each rate parameter π_j to a linear combination $\mathbf{x}^{\mathrm{T}}\boldsymbol{\beta}_j$ through $\pi_j = \exp\{\mathbf{x}^{\mathrm{T}}\boldsymbol{\beta}_j\}$, which yields a mixture of Poisson regressions:

$$f(y;\boldsymbol{\psi}_2) = \sum_{j=1}^{k} \lambda_j g(y;\mathbf{x}^{\mathrm{T}}\boldsymbol{\beta}_j),$$

  where $\boldsymbol{\psi}_2 = (\lambda_1, \ldots, \lambda_{k-1}, \boldsymbol{\beta}_1^{\mathrm{T}}, \ldots, \boldsymbol{\beta}_k^{\mathrm{T}})^{\mathrm{T}} \in \boldsymbol{\Psi}_2 \subset \mathbb{R}^{k(p+1)-1}$

Example: Earthquake Data

- Earthquakes are a natural phenomenon which happen every day
- Predicting earthquakes is nearly impossible, but studying the frequency of earthquakes that happen in a year might reveal some patterns of interest
- We study the number of strong and major earthquakes, which are earthquakes measuring at least 6.0 on the Richter scale
- We will analyze annual earthquake counts from years 1900-2021 ($n = 122$)
- Source: United States Geological Survey (2022), https://earthquake.usgs.gov/earthquakes/search/. Accessed January 24$^{\text{th}}$, 2022.

Mixed Effects Regression Mixtures

- Consider the linear mixed model (LMM)

$$\mathbf{Y}_i = \mathbf{X}_i\boldsymbol{\beta} + \mathbf{Z}_i\mathbf{u}_i + \boldsymbol{\epsilon}_i, \quad i = 1, \ldots, n, \tag{2}$$

  where $\mathbf{Y}_i \in \mathbb{R}^m$ is the response vector, $\mathbf{X}_i$ is an $m \times p$ design (incidence) matrix associated with the fixed effects, $\boldsymbol{\beta} \in \mathbb{R}^p$ is the vector of regression coefficients for the fixed effects, $\mathbf{Z}_i$ is an $m \times q$ design (incidence) matrix associated with the random effects, $\mathbf{u}_i$ is the vector of random effects, and $\boldsymbol{\epsilon}_i$ is the residual vector, where $\boldsymbol{\epsilon}_i \sim \mathcal{N}_m(\mathbf{0}, \sigma^2\mathbf{I})$
- In the classic LMM, $\mathbf{u}_i \sim \mathcal{N}_q(\mathbf{0}, \mathbf{G})$, but in the mixed effects regression mixture, we assume that $\mathbf{u}_i \sim \sum_{j=1}^{k} \lambda_j \mathcal{N}_q(\boldsymbol{\mu}_j, \mathbf{G}_j)$
- Used for clustering longitudinal (trajectory) data
- Considered the case where the error variability, σ^2, could also be different for each component
- Young and Hunter (2015) developed an expectation-conditional-maximization (ECM) algorithm for estimation and likelihood ratio tests about the variance components
- Model can also be used for response vectors of different length; i.e., m_i instead of m

Example: Infant Habituation Data

- Habituation is a psychological learning process wherein there is a decrease in response times upon repeated stimulus presentations
- Visual habituation studies in infants have attempted to predict later cognitive abilities in childhood; e.g., short visualization fixations tend to be cognitively advantaged over those with longer visualization fixations
- Hence, it is of scientific interest to investigate possible subgroups of infants based on visual habituation results as this could provide insight into their cognitive development
- Will analyze the second set of $m = 11$ trials for $n = 47$ infants at 4 months of age
- Source: H. Thomas, A. Lohaus, and H. Domsch (2011), "Extensions of Reliability Theory." In: D. R. Hunter, D. S. P. Richards, and J. L. Rosenberger (eds.) *Nonparametric Statistics and Mixture Models: A Festschrift in Honor of Thomas P. Hettmansperger*. World Scientific, Singapore, 309–316.

Determining the Number of Components

Model Selection Criteria

- Model selection criteria are widely used in selecting the correct model among a set of candidate models, this includes when trying to determine the number of components, k of a mixture model when not assumed *a priori*
  - See, for example, Chapter 6 of McLachlan and Peel (2000)
- Let $\ell(\hat{\boldsymbol{\psi}})$ be the loglikelihood of a mixture model evaluated at the maximum likelihood estimate $\hat{\boldsymbol{\psi}}$
- Letting d be the number of parameters in the given mixture model, four common information criteria are as follows:

$$\begin{aligned}
\text{AIC} &= -2\ell(\hat{\boldsymbol{\psi}}) + 2d \\
\text{BIC} &= -2\ell(\hat{\boldsymbol{\psi}}) + d\log(n) \\
\text{ICL} &= \text{BIC} + 2\left(-\sum_{i=1}^{n}\sum_{j=1}^{k}\hat{p}_{ij}\right) \\
\text{cAIC} &= -2\ell(\hat{\boldsymbol{\psi}}) + d(\log(n)+1)
\end{aligned}$$

- For multiple candidate models, we calculate an information criterion for each model and then select the model with the lowest information criterion value

Likelihood Ratio Test

- From a likelihood perspective, we consider testing

$$H_0 : k = k_0$$
$$H_A : k = k_0 + 1$$

  for some positive integer k_0

- Letting $\hat{\boldsymbol{\psi}}_1$ and $\hat{\boldsymbol{\psi}}_2$ denote the MLEs of $\boldsymbol{\psi}$ calculated under H_0 and H_A, respectively, we could consider the likelihood ratio test (LRT) statistic

$$-2 \log \Delta = 2\{\ell(\hat{\boldsymbol{\psi}}_1) - \ell(\hat{\boldsymbol{\psi}}_0)\}$$

- It is well known that standard regularity conditions do not hold in the setting of the above test, and thus the asymptotic distribution of $-2 \log \Delta$ is not the usual chi-squared distribution

Parametric Bootstrapping of the LRT

Parametric Bootstrap for the LRT Statistic

1. Fit a mixture model with k_0 and $k_0 + 1$ components to the data, $\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_n$, which leads to the EM estimates $\hat{\psi}_1$ and $\hat{\psi}_2$, respectively.
2. Calculate the (observed) loglikelihood ratio statistic. Denote this value by Ξ_{obs}.
3. Simulate a data set of size n from the null distribution (the model with k_0 components). Call this sample $\mathbf{y}_1^*, \mathbf{y}_2^*, \ldots, \mathbf{y}_n^*$.
4. Fit a mixture model with k_0 and $k_0 + 1$ components to the simulated data and calculate the corresponding bootstrap loglikelihood ratio statistic. Denote this value by Ξ^*
5. Repeat steps 3 and 4 B times to generate the bootstrap sampling distribution of the likelihood ratio statistic, $\Xi_1^*, \Xi_2^*, \ldots, \Xi_B^*$.
6. Compute the bootstrap p-value as

$$p_B = \frac{1}{B} \sum_{i=1}^{B} \mathrm{I}\{\Xi_{obs} \geq \Xi_i^*\}.$$

Example: Earthquake Data (ctd.)

- We will revisit the earthquake data and show how we arrived at $k = 6$ for the number of components for our mixture-of-Poissons model
- We will calculate BIC values for models with $k \in \{2, 3, \ldots, 6, 7\}$ components
- There is currently no direct function available in `mixtools` for computing information criteria for mixtures-of-Poissons fits, but this is an improvement that will be developed for a future release of the package

Example: Aphids Data (ctd.)

- We will revisit the aphids data and show how we arrived at $k = 2$ for the number of components for our mixture-of-regressions model
- We will calculate the AIC, BIC, ICL, and cAIC values for models with $k \in \{2, 3, 4\}$ components
- This can be done directly using the `regmixmodel.sel()` function in `mixtools`

Example: Carbon Dioxide Data (ctd.)

- We will revisit the carbon dioxide data and show how we arrived at $k = 2$ for the number of components for our mixture-of-regressions model
- We will run the parametric bootstrapping routine for successfully testing k_0 versus $k_0 + 1$ components
- When we fail to reject the null hypothesis, we then determine that value to be the number of components for our mixture model
- This can be done directly using the `boot.comp()` function in `mixtools`

Visualizing Estimated Mixture Models

Example: Hidalgo Stamp Data (ctd.)

- Output from many of the mixture model estimation routines can also directly call a plot of the fitted mixture density components (e.g., a histogram with the estimated densities for univariate data, while a scatterplot with the estimated regression relationships for a mixture of simple linear regressions) as well as a trace plot of the observed data loglikelihood
- Recently, we have updated the plotting capabilities in `mixtools` to produce `plotly`-based and `ggplot`-based graphics
- Let us return to the output for the Hidalgo stamp data and illustrate the original plotting capabilities using `base R` graphics and the new plotting capabilities using functions available from the `mixtools GitHub` repo

The Mixturegram

- Traditional numerical methods for testing the number of components in finite mixture models include the calculation of information criteria and bootstrapping approaches
- Visualizations only available for specific settings or Bayesian approaches
- Young et al. (2018) developed the **mixturegram** as a way to visualize an appropriate number of components for a finite mixture model, which can supplement the results from traditional methods or provide visual evidence when results from such methods are inconclusive
- The approach augments the data (univariate or multivariate) with posterior membership probabilities from an EM algorithm, applies an appropriate dimension reduction technique (e.g., kernel PCA or reduced-rank LDA), transforms the augmented data accordingly, and then plots the data on a set of parallel coordinates with color-coding based on an encoder that assigns each observation to a component

Example: Quasars Data (ctd.)

- We will use the updated version of the mixturegram available on `GitHub` (`plotly.mixturegram()`)
- The mixturegram requires estimated mixture fits from models with $k = 1, 2, \ldots, K$ components, where K is a practical upper limit under consideration
- One needs to code a preliminary function that enforces an identifiability constraint that is applied to each model fit; e.g., an ordering on the estimated means of the mixtures
  - See Young et al. (2018) for more details on this ordering
- We return to the quasars data where we construct the mixturegram for $k = 1, 2, \ldots, 6$ components

Final Comments

Some Other Applications Using `mixtools`

- Used to classify different seed types in maize production to understand their oil content. [Melchinger et al. (2015), "Oil Content is Superior to Oil Mass for Identification of Haploid Seeds in Maize Produced with High-Oil Inducers." *Crop Science*; 55(1):188–195.]
- Used to cluster patients with acute myeloid leukemia to aid in identifying mutations in genes in their DNA. [Kroeze et al. (2014), "Characterization of Acute Myeloid Leukemia Based on Levels of Global Hydroxymethylation." *Blood*; 124(7):1110—1118.]
- Used to evaluate the distribution of fluorodeoxyglucose (FDG) uptake in the dorsal and ventral streams in the prefrontal white matter in an effort to understand that feature in people with autism spectrum disorder and schizophrenia. [Mitelman et al. (2018), "Increased White Matter Metabolic Rates in Autism Spectrum Disorder and Schizophrenia." *Brain Imaging and Behavior*; 12(5):1290–1305.]
- Used to cluster color morph variants on manta rays based on long-term photo identification catalogs. [Venables et al. (2019), "It's Not All Black and White: Investigating Colour Polymorphism in Manta Rays Across Indo-Pacific Populations." *Proceedings of the Royal Society B*; 286(1912):1–10.]

Recent and Forthcoming Additions to `mixtools`

Recent Additions

- Launched `GitHub` repo (https://github.com/dsy109/mixtools)
- Added new graphics capabilities based on `ggplot2` and `plotly`

Forthcoming Additions

- Will launch a `Shiny` app (https://mixtools.as.uky.edu/)
- Add `S3` methods to the package, such as for prediction via the `predict()` function
- Update `boot.se()` to be fully functional with all of the mixture estimation functions, such as with the `gammamixEM()` function
- Recording a training module; make available on my website

Open Discussion

- As current (or future) end-users of the `mixtools` package, what are features that would be helpful for your research and data analysis needs?

References I

Benaglia, T., Chauveau, D., Hunter, D. R., & Young, D. S. (2009). `mixtools`: An `R` Package for Analyzing Mixture Models. *Journal of Statistical Software*, *32*(6), 1–29.

Dempster, A. P., Laird, N. M., & Rubin, D. B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society, Series B*, *39*(1), 1–38.

Hunter, D. R., & Young, D. S. (2012). Semiparametric Mixtures of Regressions. *Journal of Nonparametric Statistics*, *24*(1), 19–38.

Hurn, M., Justel, A., & Robert, C. P. (2003). Estimating Mixtures of Regressions. *Journal of Computational and Graphical Statistics*, *12*(1), 55–79.

Levine, M., Hunter, D. R., & Chauveau, D. (2011). Maximum Smoothed Likelihood for Multivariate Mixtures. *Biometrika*, *98*(2), 403–416.

McLachlan, G. J., & Peel, D. (2000). *Finite Mixture Models*. New York, NY: Wiley.

References II

Young, D. S., & Hunter, D. R. (2015). Random Effects Regression Mixtures for Analyzing Infant Habituation. *Journal of Applied Statistics*, *42*(7), 1421–1441.

Young, D. S., Ke, C., & Zeng, X. (2018). The Mixturegram: A Visualization Tool for Assessing the Number of Components in Finite Mixture Models. *Journal of Computational and Graphical Statistics*, *27*(3), 565–575.

Contact Information


derek.young@uky.edu


http://young.as.uky.edu


https://github.com/dsy109