The Journal of Open Source Software

mixR: An R package for Finite Mixture Modeling for Both Raw and Binned Data

Youjiao Yu[1]

1 Department of Statistical Science, Baylor University

DOI: 10.21105/joss.04031

Software

- Review
- Repository
- Archive

Editor: Frederick Boehm

Reviewers:

- @welch16
- @soodoku

Submitted: 29 December 2021
Published: 13 January 2022

Statement of need

R (R Core Team, 2020) provides a rich collection of packages for building and analyzing finite mixture models, which are widely used in unsupervised learning, such as model-based clustering and density estimation. For example, `mclust` (Scrucca et al., 2016) can be used to build Gaussian mixture models with different covariance structures, `mixtools` (Benaglia et al., 2010) implements parametric and non-parametric mixture models as well as mixtures of Gaussian regressions, `flexmix` (Leisch, 2004) provides a general framework for finite mixtures of regression models, `mixdist` (Macdonald et al., 2018) fits mixture models for grouped and conditional data (also called binned data). To our knowledge, almost all R packages for finite mixture models are designed to use raw data as the modeling input except `mixdist`. However, the popular model selection methods based on information criteria or bootstrapping likelihood ratio test (bLRT) (Feng & McCulloch, 1996; McLachlan, 1987; Yu & Harvill, 2019) are not implemented in `mixdist`. To bridge this gap and to unify the interface for finite mixture modeling for both raw and binned data, we implement `mixR` package that provides the following primary features.

- `mixfit()` performs maximum likelihood estimation (MLE) for finite mixture models for Gaussian, Weibull, Gamma, and Log-normal distributions via EM algorithm (Dempster et al., 1977). The model fitting is accelerated via package `Rcpp` (Eddelbuettel et al., 2011).
- `select()` selects the best model from a series of mixture models with a different number of mixture components by using Bayesian Information Criterion (BIC).
- `bs.test()` performs bLRT for two mixture models from the same distribution family but with a different number of components.

`mixR` also contains the following additional features.

- Visualization of the fitted mixture models using `ggplot2` (Wickham, 2011).
- Functions to generate random data from mixture models.
- Functions to convert parameters of Weibull and Gamma mixture models between shape-scale representation used in probability density functions and mean-variance representation which is more intuitive for people to understand the distribution.

Examples

We demonstrate how to use `mixR` for fitting finite mixture models and selecting mixture models using BIC and bLRT.

Model fitting

We fit the following four mixture models to a data set that consists of 1000 random data points generated from a Weibull mixture model with two components.

- Gaussian mixture with two components (`mod1`)
- Gaussian mixture with two components to the binned data (`mod2`)
- Gaussian mixture with three components (`mod3`)
- Weibull mixture with two components (`mod4`)

The fitted coefficients in `mod1` and `mod2` and the top two plots in Figure 1 show that binning does not cause much information loss, and we get similar fitted results using either raw data or binned data. This is usually the case when we have at least moderate data size, and the underlying mixture model is not too complex (e.g., too many mixture components). A benefit of binning is that it reduces the computation burden significantly for large data, especially when conducting bLRT, which is computationally intensive. From Figure 1 we also observe that Gaussian mixture models can provide a good fit for non-Gaussian data though the number of mixture components tends to be overestimated because more Gaussian components are needed to model the asymmetry and long tails that usually exist in non-Gaussian data.

```
library(mixR)

set.seed(101)
x <- rmixweibull(1000, c(0.4, 0.6), c(0.6, 1.3), c(0.1, 0.1))
x_binned <- bin(x, brks = seq(min(x), max(x), length = 30))

mod1 <- mixfit(x, ncomp = 2)
mod2 <- mixfit(x_binned, ncomp = 2)
mod3 <- mixfit(x, ncomp = 3)
mod4 <- mixfit(x, ncomp = 2, family = 'weibull')

mod1
## Normal mixture model with 2 components
##        comp1     comp2
## pi 0.4210604 0.5789396
## mu 0.6014690 1.3084871
## sd 0.1092375 0.0932826
##
## EM iterations: 5 AIC: -406.65 BIC: -382.11 log-likelihood: 208.32

mod2
## Normal mixture model with 2 components
##        comp1     comp2
## pi 0.4213019 0.5786981
## mu 0.6018737 1.3091224
## sd 0.1084973 0.0916267
##
## EM iterations: 9 AIC: 5813.09 BIC: 5837.63 log-likelihood: -2901.54

p1 <- plot(mod1, title = 'Gaussian Mixture (2 components)')
p2 <- plot(mod2, title = 'Gaussian Mixture (binned data 2 components)')
p3 <- plot(mod3, title = 'Gaussian Mixture (3 components)')
p4 <- plot(mod4, title = 'Weibull Mixture (2 components)')
gridExtra::grid.arrange(p1, p2, p3, p4, nrow = 2)
```

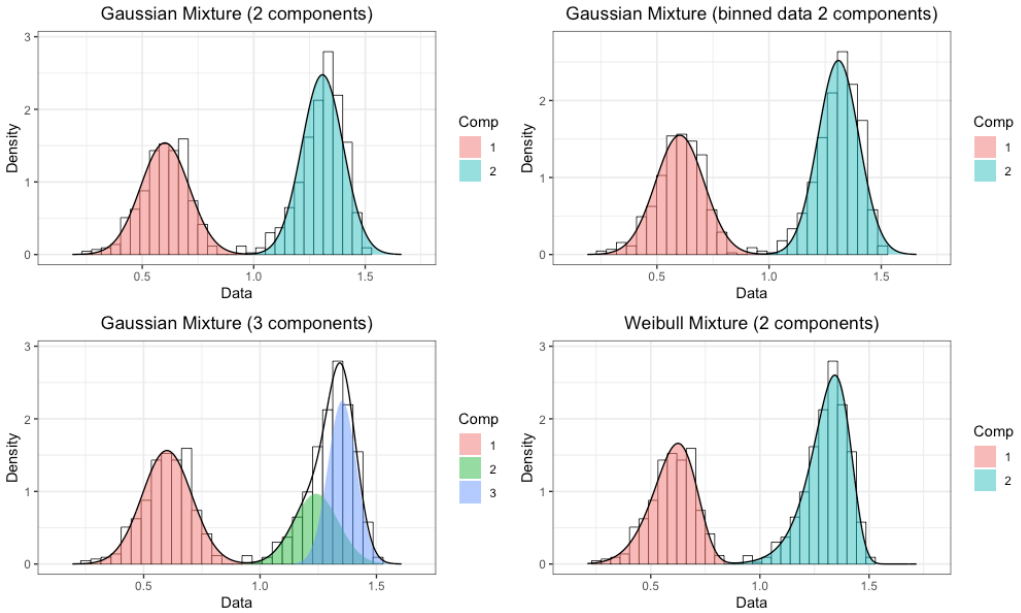

Figure 1: (top left) the fitted Gaussian mixture with two components; (top right) the fitted Gaussian mixture with two components to the binned data; (bottom left) the fitted Gaussian mixture with three components; (bottom right) the fitted Weibull mixture with two components

Model selection

Figure 2 shows that the best Gaussian mixture model selected by BIC has three components and unequal variances for each component, while the best Weibull mixture model has two components. The bLRT with $H_0 : g = 2$ versus $H_a : g = 3$ for Gaussian mixture models (using the default 100 bootstrap iterations) returns a p-value of zero, showing that Gaussian mixture with three components is significantly better than that with two components. Similarly, the same test for Weibull mixture models returns an insignificant p-value of 0.82, indicating that the Weibull mixture with three components is no better than it with two components.

```
b1 <- select(x, ncomp = 2:4)
b2 <- select(x, ncomp = 2:4, family = 'weibull')
b3 <- bs.test(x, ncomp = c(2, 3))
b4 <- bs.test(x, ncomp = c(2, 3), family = 'weibull')

b3$pvalue
## [1] 0

b4$pvalue
## [1] 0.82

par(mfrow = c(2, 2))
plot(b1)
plot(b2, main = "Weibull Mixture Model Selection by BIC")
plot(b3, main = "Bootstrap LRT for Gaussian Mixture Models\n
     (g = 2 vs. g = 3)", xlab = 'Bootstrap Test Statistics')
plot(b4, main = "Bootstrap LRT for Weibull Mixture Models\n
     (g = 2 vs. g = 3)", xlab = 'Bootstrap Test Statistics')
```

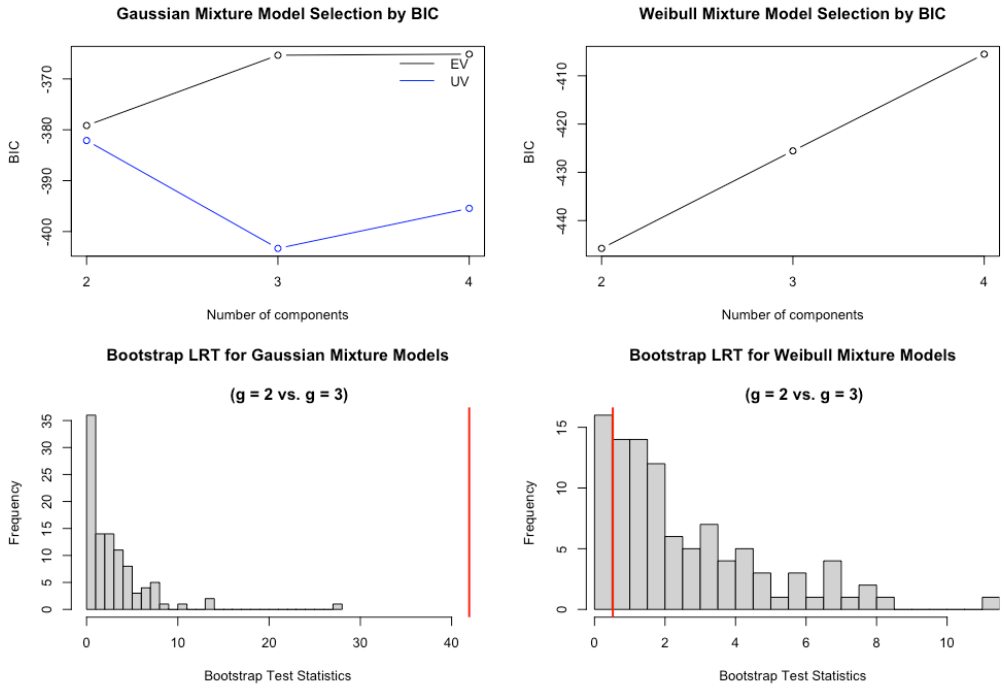

Figure 2: (top left) Gaussian mixture model selection using BIC (UV stands for unequal variances for each mixture components and EV stands for equal variance); (top right) Weibull mixture model selection using BIC; (bottom left) bLRT with $H_0 : g = 2$ versus $H_a : g = 3$ for Gaussian mixture models; (bottom right) bLRT with $H_0 : g = 2$ versus $H_a : g = 3$ for Weibull mixture models

Summary

`mixR` unifies the interface for fitting and comparing finite mixture models for both raw data and binned data for distributions including Gaussian, Weibull, Gamma, and Log-normal. The package also provides features for generating random data from mixture models, conversion of parameters for Weibull and Gamma models, and model visualization in `ggplot2`. The heavy computation in `mixR` is completed in C++ using `Rcpp`.

`mixR` is actively used by researchers and practitioners in various fields (Buckland et al., 2021; Büchel & Corman, 2021; Jung et al., 2020; Korne et al., 2021; Ogana, 2020; Sylvestre et al., 2020; S.-I. Yang et al., 2021; Z. R. Yang, 2021).

References

Benaglia, T., Chauveau, D., Hunter, D. R., & Young, D. S. (2010). mixtools: An R package for analyzing mixture models. *Journal of Statistical Software*, *32*, 1–29. https://doi.org/10.18637/jss.v032.i06

Buckland, H. M., Saxby, J., Roche, M., Meredith, P., Rust, A. C., Cashman, K. V., & Engwell, S. L. (2021). Measuring the size of non-spherical particles and the implications for grain size analysis in volcanology. *Journal of Volcanology and Geothermal Research*, *415*, 107257. https://doi.org/10.1016/j.jvolgeores.2021.107257

Büchel, B., & Corman, F. (2021). Modeling conditional dependencies for bus travel time estimation. *Physica A: Statistical Mechanics and Its Applications*, 126764. https://doi.

org/10.1016/j.physa.2021.126764

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 (Methodological)*, *39*(1), 1–22. https://doi.org/10.1111/j.2517-6161.1977.tb01600.x

Eddelbuettel, D., François, R., Allaire, J., Ushey, K., Kou, Q., Russel, N., Chambers, J., & Bates, D. (2011). Rcpp: Seamless R and C++ integration. *Journal of Statistical Software*, *40*(8), 1–18. https://doi.org/10.18637/jss.v040.i08

Feng, Z. D., & McCulloch, C. E. (1996). Using bootstrap likelihood ratios in finite mixture models. *Journal of the Royal Statistical Society: Series B (Methodological)*, *58*(3), 609–617. https://doi.org/10.1111/j.2517-6161.1996.tb02104.x

Jung, H., Senf, C., Jordan, P., & Krueger, T. (2020). Benchmarking inference methods for water quality monitoring and status classification. *Environmental Monitoring and Assessment*, *192*(4), 1–17. https://doi.org/10.1007/s10661-020-8223-4

Korne, C. M. de, Winkel, B. M., Oosterom, M. N. van, Chevalley-Maurel, S., Houwing, H., Sijtsma, J. C., Azargoshasb, S., Baalbergen, E., Franke-Fayard, B., Leeuwen, F. van, & others. (2021). Clustering and erratic movement patterns of syringe-injected versus mosquito-inoculated malaria sporozoites underlie decreased infectivity. *Msphere*, *6*(2), e00218–21. https://doi.org/10.1128/mSphere.00218-21

Leisch, F. (2004). *FlexMix: A general framework for finite mixture models and latent glass regression in R*. https://doi.org/10.18637/jss.v011.i08

Macdonald, P., Du, J., & Macdonald, M. P. (2018). Package 'mixdist.' *Version 0.5–5*.

McLachlan, G. J. (1987). On bootstrapping the likelihood ratio test statistic for the number of components in a Normal mixture. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, *36*(3), 318–324. https://doi.org/10.2307/2347790

Ogana, F. N. (2020). Does the inclusion of truncation point in a finite mixture model improve diameter distribution estimation of degraded stand? *Journal of Sustainable Forestry*, 1–15. https://doi.org/10.1080/10549811.2020.1841009

R Core Team. (2020). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Scrucca, L., Fop, M., Murphy, T. B., & Raftery, A. E. (2016). mclust 5: Clustering, classification and density estimation using Gaussian finite mixture models. *The R Journal*, *8*(1), 289. https://doi.org/10.32614/rj-2016-021

Sylvestre, É., Burnet, J.-B., Smeets, P., Medema, G., Prévost, M., & Dorner, S. (2020). Can routine monitoring of E. coli fully account for peak event concentrations at drinking water intakes in agricultural and urban rivers? *Water Research*, *170*, 115369. https://doi.org/10.1016/j.watres.2019.115369

Wickham, H. (2011). ggplot2. *Wiley Interdisciplinary Reviews: Computational Statistics*, *3*(2), 180–185. https://doi.org/10.1002/wics.147

Yang, S.-I., Cao, Q. V., Shoch, D. T., & Johnson, T. (2021). Characterizing stand and biomass tables from diameter distribution models: A case study for mixed-hardwood forests in eastern Tennessee, USA. *Forest Science*. https://doi.org/10.1093/forsci/fxab051

Yang, Z. R. (2021). *Biological pattern discovery with R: Machine learning approaches*. World Scientific. https://doi.org/10.1142/12366

Yu, Y., & Harvill, J. L. (2019). Bootstrap likelihood ratio test for Weibull mixture models fitted to grouped data. *Communications in Statistics-Theory and Methods*, *48*(18), 4550–4568. https://doi.org/10.1080/03610926.2018.1494838