August 12, 2020

Dear Catherine Hurley,

Re: Re-submission of article on rmdcev package

Thank you for organizing the reviews and the opportunity to re-submit the manuscript for publication in the R Journal. I have responded point-by-point to the reviewers below. I have tried to be concise in the changes to the manuscript as the paper is already quite long. When warranted, I have added additional references for further reading. In addition to changes in response to the constructive reviewer comments, I have also made some changes to the manuscript to reflect changes to the package since the initial submission of the paper in November 2019. Most of these changes have focused on expanding the capabilities of the package and providing more user friendly error messages. The revised package is uploaded to CRAN.

Sincerely,

Patrick Lloyd-Smith
Assistant Professor
Department of Agricultural and Resource Economics
Global Institute for Water Security
University of Saskatchewan
101-121 Research Drive
Saskatoon, SK S7N 1K2

Reviewer 1

1 Overall Feedback

I have read the paper submitted to the R Journal with interest and believe that the package is a welcome and useful addition to the R-packages already available in the domain of choice modeling, such as "gmnl", "mlogit", "apollo" and "mnlogit" package.

Response: Thank you for the substantive and helpful comments on the package and the manuscript. I have provided detailed responses to your comments below.

2 Comments on the article

1. Introduction: I'd suggest explaining what are the main models estimated by rmdcev and give an intuitive explanation in which case use them for the practitioners. In the posterior sections, you can give a more detail explanation of each model.

Response: Thank you for the suggestion and I agree that explaining these models early on is helpful. I have now clarified between the two main model specifications estimated by rmdcev (MDCEV and the KT models in environmental economics), the two extensions to incorporate preference heterogeneity (latent class and random parameters), and the two estimation algorithms (MLE and Bayesian estimation).

2. The version of the package used in the manuscript should be explicitly stated in the Introduction.

Response: I have now included the specific package version in the first footnote of the paper.

3. A table summarizing the most important functions in the package would be useful, possible along with a listing model that can be fitted.

Response: I have revised the introduction to clarify the model specifications that can be estimated and identify the two main functions in rmdcev (mdcev and mdcev.sim). I created several potential tables to summarize these functions, but did not find one that clarified the models options beyond the revised text. The main challenge I had was that there are really only two overall model specifications and therefore I focused on improving the flow of the text rather than adding a table.

4. Data: Please, provide an example on how the data must be arrange when facing alternative-specific attributes.

Response: I have revised the data and formula section to better articulate how the alternative-specific attributes should be included in the data and formula.

5. Arguments of the function: there is a typo in the formula argument.

Response: Thanks for finding this typo. Fixed now.

6. I think that the package is **Formula**.

Response: Yes, you are correct. I have revised the text accordingly.

7. In my experience, many users have problems interpreting the results. Given this, I'd suggest providing a short paragraph interpreting the parameters estimates after each application. It also helps experienced users to confirm that they have understood the output correctly.

Response: This is an excellent suggestion and I have expanded the discussion and interpretation of results. I focussed the discussion on the earlier models to help interpretation but avoid repeating myself for the later models. I have also provided additional references of papers that provide useful empirical application of these models.

8. As a suggestion, an additional Section explaining the typical convergence problems of the ML and potential solutions would be very useful.

Response: I have added a brief section on computational and estimation issues following the estimation section.

9. Include comma or period at the end of each Equation.

Response: Comments or periods have been added to each equation.

3 Comments on the package

1. The LC model is very sensitive to initial values. How are they computed?

Response: This is a good point and something that is not clear in the previous version of the package. The user has the option to specify starting values through the initial.parameter argument of mdcev. If initial parameter values are not provided, the default for the LC model is to use the MLE parameter estimates from the single class model and then slightly perturb the psi and gamma coefficients for the classes to ensure the same starting values aren't used for both classes. The code to compute these perturbations for psi is provided as an example below

Code excerpt from Lines 54 to 58 of maxLikeMDCEV

```
# add shift to psi values values
init.psi <- init.par$psi
init.shift <- seq(-0.02, 0.02, length.out = stan_data$NPsi)
for (i in 1:stan_data$NPsi)
        init.psi[i] <- init.psi[i] + init.shift[i]
```

2. Based on my previous comment, the user should be able to include their own initial values. A start argument is needed.

Response: I have added the functionality that the user can set their own initial parameter values for the LC model. I have also added an example to show how the list of initial parameter values need to be set. I have revised the initial.parameter argument to describe these details.

```
initial.parameters: The default for fixed and random parameter specifications is to use random starting values.
#' For LC models, the default is to use slightly adjusted MLE point estimates from the single class model.
#' Initial parameter values should be included in a named list. For the "hybrid" specification,
#' initial parameters can be specified as:
#' inititial.parameters = list(psi = array(runif(1), dim = c(K, num_psi)),
#'                      gamma = array(1, dim = c(K, num_alt)),
#'                      alpha = array(0.5, dim = c(K, 0)),
#'                      scale = array(1, dim = c(K)))
#' where K is the number of classes (i.e. K = 1 is used for single class models),
#' num_psi is number of psi parameters, and num_alt is number of non-numeraire alternatives.
```

3. If random _parameters = "cor", is there any function to retrieve the standard deviation and their standard errors? If not, a better object oriented programming could be employed using cov() as in mlogit and gmnl package.

Response: Yes, parameter estimates for standard deviations and associated can be retrieved using the summary function. I have added a note after the model summary detailing how the full covariance matrix can be retrieved.

The full covariance matrix can be accessed using the print(output$stan_fit, pars = 'Sigma') command where output is name of model output"

Reviewer 2

The manuscript entitled *Multiple Discrete-Continuous Extreme Value Model Estimation and Simulation in R: The rmdcev Package* introduces the *rmdcev* R package, summarising the theoretical model it focuses on (Bhat 2008's MDCEV), as well as including multiple examples on how to use the package.

The manuscript is well written and easy to follow. The package provides a fast and accessible way to estimate MDCEV models. These models have been around since 2008 (and similar ones since even further back), but its adoption has been slow non the least due to a lack of convenient estimation software. The package presented in the manuscript has the potential to bridge that gap.

However, before recommending the manuscript for publication, there are a few aspects that would benefit for further discussion or explanation. These are the following.

- What correction for truncated data is implemented in rmdcev? Please explain.

Response: I have added additional information to the trunc_data argument of mdcev as follows:

Whether the estimation should be adjusted for truncation of non-numeraire alternatives. This option is useful if the data only includes individuals with positive non-numeraire consumption levels such as recreation data collected on-site. To account for the truncation of consumption, the likelihood is normalized by one minus the likelihood of observing zero consumption for all non-numeraire alternatives (i.e. likelihood of positive consumption) following Englin, Boxall and Watson (1998) and von Haefen (2003).

- About the gamma parameter. Can the gamma_k be parametrised in the same way that psi_k can?

Response: The current package version only allows additional parameters in the psi_k term. Given that very few empirical applications that I am aware of incorporate alternative or individual-specific variables in gamma_k, I'd prefer to keep the simpler model specifications. I have now noted this explicitly in the package documentation and manuscript.

Are gamma_k or exp(gamma_k) reported? Seems like exp(gamma_k) is reported (hence the use of the delta method for standard errors), but it is not mentioned explicitly.

Response: Yes, the transformed parameters exp(gamma*_k) is reported rather than gamma*_k. I have clarified this distinction in the manuscript.

> Finally, could the estimated gamma parameters include the name of the alternative, just as the psi parameters do in the example?

Response: I have revised the summary output so that the gamma and alpha parameters return the alternative names instead of simple numbers when they are alternative-specific.

- Concerning the flat_priors option, it is not clear to me how this argument influences Maximum Likelihood Estimation (MLE). How could the optimising function not be equal to the loglikelihood?

Response: I have revised the function argument to clarify the relationship between MLE and flat_prior.

flat_priors indicator if completely uninformative priors should be specified. Defaults to 1 if MLE used and 0 if Bayes used. If using MLE and set flat_priors = 0, penalized MLE is used and the optimizing objective is augmented with the priors.

- Does MLE use multithreading? More in general (also including Bayesian estimation), if openMP is used for providing multithreading at the level of C++ code, then the code should consider the posibility of the system not supporting openMP (chiefly mac OS users). If the package uses openMP, is it protected for for Mac users? (see https://ankargren.github.io/avoiding-openmp-problems-in-rcpparmadillo-dependentpackages-on-os-x)

Response: The current version of the package does not use multithreading and I hope to implement this feature in the future. Multithreading support is currently being implemented throughout the Stan software ecosystem but this is not easily implemented in the current version of the rstan package. I look to add this feature once it becomes available in rstan.

- Looking at the output of bayesian estimation, it looks like a different thread (worker) I used for each chain. What is the relation between number of chains and number of cores?

Response: I have updated the argument descriptions to clarify the differences

n_chains: The number of independent Markov chains in Bayesian estimation.

n_cores: The number of cores used to execute the Markov chains in parellel in Bayesian estimation. Can set using options(mc.cores = parallel::detectCores()).

- Could you briefly explain how the n_iterations parameter relates to the number of warm-up and post-warm-up draws in the chain?

Response: I have included a sentence clarifying this point in the n_iterations argument explanation

n_iterations: The number of iterations to use in Bayesian estimation. The default is for the number of iterations to be split evenly between warmup and posterior draws. The number of warmup draws can be directly controlled using the warmup argument (see rstan::sampling)

- I would recommend the delta method as the default for calculating the s.e., as using draws to calculate them can mask issues if too few draws are used.

Response: I agree that a low number of draws can be problematic for inference. I have switched the default to "deltamethod" and have included more explicit instructions that draws are required if the user wants to incorporate parameter uncertainty for demand and welfare simulations.

- Rmdec uses the results of a regular MDCEV model as starting values for a latent class model. But latent class models require different starting values for each class, otherwise optimisation is likely to follow the same path for both classes. How does rmdcev deals with this?

Response: This is a good point and something that is not clear in the previous version of the package. The user has the option to specify starting values through the initial.parameter argument of mdcev. If initial parameter values are not provided, the default for the LC model is to use the MLE parameter estimates from the single class model and then slightly perturb the psi coefficients for the classes to ensure the same starting values aren't used for both classes. This approach is similar to the approach implemented in the gmnl package.
The code to compute these perturbations is provided below for reference.

Code excerpt from Lines 54 to 58 of maxLikeMDCEV

```
# add shift to psi values values
init.psi <- init.par$psi
init.shift <- seq(-0.02, 0.02, length.out = stan_data$NPsi)
for (i in 1:stan_data$NPsi)
        init.psi[i] <- init.psi[i] + init.shift[i]
```

I have revised the initial.parameter argument to describe these details and in the manuscript and provided an example.

```
initial.parameters: The default for fixed and random parameter specifications is to use random starting values.
#' For LC models, the default is to use slightly adjusted MLE point estimates from the single class model.
#' Initial parameter values should be included in a named list. For example, for the "hybrid" specification,
#' initial parameters can be specified as:
#' inititial.parameters = list(psi = array(0, dim = c(K, num_psi)),
#'                    gamma = array(1, dim = c(K, num_alt)),
```

#'   alpha = array(0.5, dim = c(K, 1)),
#'   scale = array(1, dim = c(K)))
#' where K is the number of classes (i.e. K = 1 is used for single class models),
#' num_psi is number of psi parameters, and num_alt is number of non-numeraire alternatives.

- My understanding of the difference between conditional and unconditional draws is as follows. While conditional draws are faster they require consumed amounts to be observed. The unconditional draws, instead do not require this. Therefore, the conditional draws are better suited for scenario evaluation, where we compare new situations to an observed baseline. The unconditional draws, instead, are better suited for out-of-sample forecasting. If correct, this might be a useful difference to mention in the manuscript.

Response: Yes, this is part of the rationale for using conditional versus unconditional draws and I have expanded the discussion in this section.

The main differences between these two error draw approaches is that in the conditional approach, errors are drawn such that the model perfectly predicts the observed consumption patterns in the baseline state (von Haefen and Phaneuf, 2005). The conditional approach uses observed behaviour by individuals to characterize unobserved heterogeneity and can be useful for scenario simulation as the baseline matches observed behavior. This is especially true if poor in-sample behavioral predictions is found using the unconditional approach (von Haefen, 2003). The unconditional approach draws all errors based on distributional assumptions and is necessary for out-of-sample forecasting. If the model correctly specifies the data generating process, the sample means of the conditional and unconditional approaches should converge in expectation. Another difference between the two approaches is that the unconditional approach uses more computation time as there is a need to calculate consumption patterns in the baseline state as well as simulate the entire distribution of unobserved heterogeneity.

Minor comments

- Please mention in the introduction that *rmdcev* assumes inclusion of an outside good.

Response: I have included the following sentence in the introduction

"that rmdcev…4) only estimates model specifications with an outside good that is always consumed whereas \pkg{apollo} can estimate models without an outside good."

- "The multiple-discrete continuous extreme value (MDCEV) demand model, also commonly called Kuhn-Tucker (KT) models in economics..." This is not very precise, the MDCEV model is a particular case of a KT model.

Response: I have revised the introduction to clarify the differences between these two terms. I have also added the KT specifications used in the environmental economics literature and distinguished this from the MDCEV specification.

- I would recommend writing equation (1) in two lines: one line for max U, and another for the s.t. …, to ease reading.

Response: I have revised the equation to cover two lines

- In equation (1), "y" does not need to be annual income, but any arbitrarily defined budget.

Response: Good point. I have revised the sentence to generalize the notion of a budget constraint beyond income.

- In page 3, after equation (5), it says gamma_k = exp(gamma_k), I would recommend a slight change in notation to gamma_k = exp(gamma'_k), where the gamma'_k is the parameter really estimated, but exp(gamma'_k) is reported to the user (if that is the case). Similarly for alpha_k.

Response: I have made the suggested revisions and have included a sentence clarifying this point in the text.

- After equation 9, the author state "However, this fixed MDCEV specification is quite restrictive as it imposes that all individuals have the same tastes for altenatives (i.e. preference homeogeneity)". This is not exactly true, as z_k can include characteristic of the individual, allowing for systematic taste variations (e.g. older individuals have a higher base utility for a given alternative).

Response: I have revised this text as follows

However, this fixed MDCEV specification is quite restrictive as it can only incorporate preference heterogeneity through interaction terms with observed individual characteristics. Without these interaction terms, the fixed MDCEV specification imposes the assumption that all individuals have the same tastes for alternatives (i.e. preference homogeneity). This assumption is relaxed in the next two specifications which are able to accommodate both observed and unobserved preference heterogeneity.

- In the text under the title "Random parameters (RP-MDCEV) models", the word "flexible" is misspelled as "fleixlbe".

Response: Thanks for finding this typo. Fixed now.

- Please briefly explain or provide a reference for the "label switching" problem.

Response: For space considerations I have provided a reference to the "label switching" problem (Jasra et al. (2005)).

- In page 4, it says "in additional", it should say "in addition".

Response: Thanks for finding this typo. Fixed now.

- The piece of code summarising the data “data_rec %>% group_by(alt) %>% ...” uses tidyverse notation. I would recommend either mentioning the necessary packages to run this code, or providing code using only base R functions.

Response: I have converted this code to use base R functionality.

- Instead of asking the user to order the data by id and alternative, why not have mdcev.data do it for the user?

Response: The package has been revised and mdcev.data will now arrange the data by id and alternative. A message is provided to users to tell them the data is being sorted in this manner.

- There is a formatting issue in page six, with the text “\code{formula**:”

Response: Thanks for finding this typo. Fixed now.

- About the constant in the formula. Is that constant the same for all non-numeraire alternatives or is it alternative specific? If I want to include alternative-specific attributes, then these should have the value zero for all other alternatives or is there a more efficient way to include them?

Response: I have revised the formula arguments to clarify how alternative-specific variables and alternative-specific constants are handled. The formula is now for alternative-specific variables and alternative-specific constants for the psi_k parameters can be included using the “psi_ascs” argument. Alternative-specific attributes can be included as a single column of the data.

- Installation takes quite some time, due to all the compiling. A warning to the reader may be welcomed.

Response: I have revised the package to cut down on the compilation times and now the models are pre-compiled so installation time should be very quick.

- When using the prior_psi_sd and prior_gamma_sd, are these s.d. assumed for all parameters in all alternatives?

Response: Yes, there is currently not the option to set alternative-specific priors.

- When estimating an MDCEV without random parameters using Bayesian techniques. Are the the reported s.e. calculated as the s.d. of each parameters’ chain?

Response: Yes, the s.e’s are actually the standard deviations across all posterior draws. I have changed the mdcev.summary command to specify that these are standard deviations rather than standard errors.

- In page 14 it says “The γ parameters, on the other hand, show that satiation between classes is quite different between the classes”. It should probably say “The γ parameters, on the other hand, show that satiation between classes is quite different”.

Response: Thanks for finding this typo. Fixed now.

- In page 16 it says “scenaros”, it should say “scenarios”.

Response: Thanks for finding this typo. Fixed now.