An R Markdown program to analyse responses to a Discrete Choice Experiment (DCE) survey exploring the online help seeking preferences of socially anxious young people

Apply Choice Models

Matthew P Hamilton[1,*]

27 October 2022

[1] Orygen, Parkville, Australia

[*] Correspondence: Matthew P Hamilton <matthew.hamilton@orygen.org.au>

Suggested citation: "Matthew Hamilton (2022). dce_sa_analysis: An R Markdown program to analyse responses to a Discrete Choice Experiment (DCE) survey exploring online help seeking in socially anxious young people. Zenodo. https://doi.org/10.5281/zenodo.7223286. Version 0.0.9.1"

1 About this code

1.1 Purpose

This program makes predictions using models derived from a Discrete Choice Experiment study that is currently being written up. Future versions of this program will include details of the parent study.

The predictions generated by this program can be used to predict willingness to pay values and market shares of alternative hypothetical social anxiety apps.

1.2 Status

This code is the same version as was used to analyse study data. The only items that have been modified are those that remove references to the local directory structure on which study data was stored.

1.3 Use

This code can be run either by "knitting" the parent RMD file or by manually executing each code chunk. The code as is currently optimised for being knit in one step, which means that in a number of instance the value "Y" has been supplied to the `consent_1L_chr` argument of program functions. Supplying this value overrides the default behaviour of functions that write files which is to prompt users for their active consent prior to to write files and directories to their machine. If running this code interactively, we recommend omitting the `consent_1L_chr` argument as this will provide you with greater transparency about what files and directories are being written to your machine.

2 Prepare workspace

2.1 Install and load required libraries

If you do not already have the required libraries to run this program installed, you can do so by un-commenting and running the following lines.

```
# devtools::install_github("ready4-dev/ready4")
# devtools::install_github("ready4-dev/mychoice") # add lwgeom to imports
```

Next we load the libraries required to run this program.

```
library(ready4)
library(mychoice)
library(ready4use)
```

2.2 Specify data directories

We begin by specifying where our input data can be located and where we wish to write our outputs to. You must supply these details or the rest of this code will not work.

```
paths_ls <- list(input_data_dir_1L_chr = "PROVIDE DETAILS HERE",
                 output_data_dir_1L_chr = "PROVIDE DETAILS HERE",
                 raw_data_fl_nms_chr = "PROVIDE DETAILS HERE")
```

2.3 Reproducibility

We now set a seed to aid reproducibility.

```
set.seed(1001)
```

Having set the seed, it is now likely that if you run the syntax described in this document on your own installation of R you will get identical results to those reported in this document. However, if you do not, it may be that you have a different version of R, or of some of the packages that we used to produce this analysis. We therefore save a record of the software that we have on the machine used for this analysis so this can be made available for comparisons.

```
session_ls <- sessionInfo()
```

3 Create custom functions

We now create a number of functions that we will use in subsequent parts of this program.

3.1 Ingest additional data

4 Apply choice models

4.1 Calculate Willingness To Pay

Our next step is to calculate Willingness To Pay values for the different attributes of social anxiety apps. We can do this both from utililty space or from willingness to pay space.

```r
analysis_ls <- add_analysis(mdls_ls = mdls_ls,
                            records_ls = records_ls,
                            what_chr = "wtp")
```

4.2 Predict market share

A core motivation for the Entourage DCE was to explore the potential viability of a user-pays financing model for a future roll-out of the Entourage App. This concluding section of the report aims to address this issue by using the models previously estimated to predict what market share Entourage could secure compared to a competing app representative of current offerings and not using an app at all. We do this using two alternative approaches. The first approach is based on the (strong) assumption that our sample is representative of our target market and therefore ignores respondent characteristics. The second includes individual characteristic predictors to make predictions that account for the characteristics of our target market.

4.2.1 Predict from existing choice situations

We will use the multinomial logit model we estimated earlier to make predictions about the market share of each option in a given choice situation, assuming that the preferences from our sample are representative of those of our target market.

The first set of predictions we make are of the market shares of the choice situations presented to survey respondents.

```r
analysis_ls <- add_analysis(mdls_ls,
                            analysis_ls = analysis_ls,
                            dce_design_ls = dce_design_ls,
                            records_ls = records_ls,
                            what_chr = "share")
```

4.2.2 Predict from custom choice situations

The previous step had the significant limitation of predictions being confined to choice situations that were included in the survey. To explore the situations that might apply in the roll-out of the Entourage app we need to generate some new choice situations. Our first step is to generate a choice situation of the Entourage app as it is currently configured compared to an app representative of current market offerings and a no-choice option. We can then predict the potential market shares of the options in the new hypothetical choice situation. We make predictions using both the multinomial logit and mixed logit models.

```r
analysis_ls <- add_new_choice_cmprsn(analysis_ls,
                                     dce_design_ls = dce_design_ls,
                                     mdl_params_ls = mdl_params_ls,
                                     new_choices_ls = list(list(Cost = 25, Endorsers = "expert", Information_sharing = NA_character_,
                                                                Outcomes = NA_character_, Social = "peer_clinician_mod"),
                                                           list(Cost = 0, Endorsers = NA_character_, Information_sharing = NA_character_,
                                                                Outcomes = NA_character_, Social = NA_character_)),
                                     records_ls = records_ls,
                                     altv_nms_chr = c("Entourage", "Competing App", "No App"),
                                     with_chr = c("mnl_mlogit_mdl","mixl_mlogit_mdl"))
```

We can then see how the predicted market share changes in response to variations in this choice situation (in the following example, we add a user controlled information sharing setting to the Entourage app, which results in an extra 6% of predicted market share).

```r
analysis_ls <- add_new_choice_cmprsn(analysis_ls,
                                     dce_design_ls = dce_design_ls,
                                     mdl_params_ls = mdl_params_ls,
                                     new_choices_ls = list(list(Cost = 25, Endorsers = "expert", Information_sharing = "user_settings",
                                                                Outcomes = NA_character_, Social = "peer_clinician_mod"),
                                                           list(Cost = 0, Endorsers = NA_character_, Information_sharing = NA_character_,
                                                                Outcomes = NA_character_, Social = NA_character_)),
                                     records_ls = records_ls,
                                     altv_nms_chr = c("Entourage", "Competing App", "No App"),
                                     with_chr = c("mnl_mlogit_mdl","mixl_mlogit_mdl"))
```

We can do this iteratively for a range of values, for example by varying the price attribute of the Entourage app.

```r
analysis_ls <- add_cost_comparison(analysis_ls,
                                   choices_ls = analysis_ls$new_data_ls$comparison_1_ls$new_choices_ls,
                                   dce_design_ls = dce_design_ls,
                                    mdls_ls = mdls_ls,
                                    mdl_params_ls = mdl_params_ls,
                                    records_ls = records_ls,
                                    cost_range_dbl = 25:50,
                                    altv_nms_chr = c("Entourage", "Competing App", "No App"),
                                    altv_to_modify_1L_int = 1,
                                    with_chr = c("mnl_mlogit_mdl","mixl_mlogit_mdl"))
```

5 Write output files

We now save our output. If running this code interactively, we recommend omitting the `consent_1L_chr` argument.

```
write_obj_with_prompt(analysis_ls,
                      obj_nm_1L_chr = "analysis_ls",
                      outp_dir_1L_chr = paths_ls$Results,
                      consent_1L_chr = "Y")
```