Package 'ramchoice'

October 23, 2021

Type Package

Title Revealed Preference and Attention Analysis in Random Limited Attention Models

Description It is widely documented in psychology, economics and other disciplines that socio-economic agent may not pay full attention to all available alternatives, rendering standard revealed preference theory invalid. This package implements the estimation and inference procedures of Cattaneo, Ma, Masatlioglu and Suleymanov (2020) <arXiv:1712.03448> and Cattaneo, Cheung, Ma, and Masatlioglu (2021) <arXiv:2110.10650>, which utilizes standard choice data to partially identify and estimate a decision maker's preference and attention. For inference, several simulation-based critical values are provided.

Version 2.0

Author Matias D. Cattaneo, Paul Cheung, Xinwei Ma, Yusufcan Masatlioglu, Elchin Suleymanov

Maintainer Xinwei Ma <x1ma@ucsd.edu>

Imports MASS

Depends R (>= 3.1.0)

License GPL-2

Encoding UTF-8

RoxygenNote 7.1.2

NeedsCompilation no

Repository CRAN

Date/Publication 2021-10-23 00:20:02 UTC

R topics documented:

`ramchoice-package` *ramchoice: Revealed Preference and Attention Analysis in Random Limited Attention Models*

Description

Information about socio-economic agent's preference (consumer, firm, organization, voter, etc.) is important not only for understanding the decision-making process, but also for conducting welfare analysis and providing robust policy recommendations. However, it is widely documented in psychology, economics and other disciplines that decision makers may not pay full attention to all available alternatives, rendering standard revealed preference theory invalid.

This package implements the estimation and inference procedures documented in Cattaneo, Ma, Masatlioglu, and Suleymanov (2020), and Cattaneo, Cheung, Ma, and Masatlioglu (2021), which utilize standard choice data to partially identify decision maker's preference and attention. For statistical inference, several simulation-based critical values are provided.

The following functions are provided: `revealPref` and `revealAtte` (the main functions for revealed preference and attention analysis), `sumData`, `genMat`, `logitAtte`, `logitSimu`. A simulated dataset `ramdata` is also included for illustration purposes.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

genMat *Generate Constraint Matrices*

Description

genMat generates constraint matrices for a range of preference orderings according to (i) the monotonic attention assumption proposed by Cattaneo, Ma, Masatlioglu, and Suleymanov (2020), (ii) the attention overload assumption proposed by Cattaneo, Cheung, Ma, and Masatlioglu (2021), and (iii) the attentive-at-binaries restriction.

This function is embedded in revealPref.

Usage

```
genMat(
  sumMenu,
  sumMsize,
  pref_list = NULL,
  RAM = TRUE,
  AOM = TRUE,
  limDataCorr = TRUE,
  attBinary = 1
)
```

Arguments

sumMenu	Numeric matrix, summary of choice problems, returned by sumData.
sumMsize	Numeric matrix, summary of choice problem sizes, returned by sumData.
pref_list	Numeric matrix, each row corresponds to one preference. For example, c(2,3,1) means 2 is preferred to 3 and to 1. When set to NULL, the default, c(1,2,3,...), will be used.
RAM	Boolean, whether the restrictions implied by the random attention model of Cattaneo, Ma, Masatlioglu, and Suleymanov (2020) should be incorporated, that is, their monotonic attention assumption (default is TRUE).
AOM	Boolean, whether the restrictions implied by the attention overload model of Cattaneo, Cheung, Ma, and Masatlioglu (2021) should be incorporated, that is, their attention overload assumption (default is TRUE).
limDataCorr	Boolean, whether assuming limited data (default is TRUE). When set to FALSE, will assume all choice problems are observed. This option only applies when RAM is set to TRUE.
attBinary	Numeric, between 1/2 and 1 (default is 1), whether additional restrictions (on the attention rule) should be imposed for binary choice problems (i.e., attentive at binaries).

Value

R	Matrices of constraints, stacked vertically.
ConstN	The number of constraints for each preference, used to extract from R individual matrices of constraints.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

Examples

```
# Load data
data(ramdata)

# Generate summary statistics
summaryStats <- sumData(ramdata$menu, ramdata$choice)

# Generate constraint matrices
constraints <- genMat(summaryStats$sumMenu, summaryStats$sumMsize)
constraints$ConstN
constraints$R[1:10, 1:10]
```

---

logitAtte — *Compute Choice Probabilities and Attention Frequencies for the Logit Attention Rule*

Description

logitAtte computes choice probabilities and attention frequencies for the logit attention rule considered by Brady and Rehbeck (2016). To be specific, for a choice problem S and its subset T, the attention that T attracts is assumed to be proportional to its size: |T|^a, where a is a parameter that one can specify. It will be assumed that the first alternative is the most preferred, and that the last alternative is the least preferred.

This function is useful for replicating the simulation results in Cattaneo, Ma, Masatlioglu, and Suleymanov (2020), and Cattaneo, Cheung, Ma, and Masatlioglu (2021).

Usage

```
logitAtte(mSize = NULL, a = NULL)
```

Arguments

mSize	Positive integer, size of the choice problem.
a	Numeric, the parameter of the logit attention rule.

Value

choiceProb	The vector of choice probabilities.
atteFreq	The attention frequency.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

R. L. Brady and J. Rehbeck (2016). Menu-Dependent Stochastic Feasibility. *Econometrica* 84(3): 1203-1223. doi: 10.3982/ECTA12694

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

Examples

```
logitAtte(mSize = 5, a = 2)
```

---

logitSimu *Choice Data Simulation Following the Logit Attention Rule*

Description

logitSimu simulates choice data according to the logit attention rule considered by Brady and Rehbeck (2016). To be specific, for a choice problem S and its subset T, the attention that T attracts is assumed to be proportional to its size: |T|^a, where a is a parameter that one can specify. It will be assumed that the first alternative is the most preferred, and that the last alternative is the least preferred.

This function is useful for replicating the simulation results in Cattaneo, Ma, Masatlioglu, and Suleymanov (2020), and Cattaneo, Cheung, Ma, and Masatlioglu (2021).

Usage

```
logitSimu(n, uSize, mSize, a)
```

Arguments

n	Positive integer, the effective sample size for each choice problem.
uSize	Positive integer, total number of alternatives.
mSize	Positive integer, size of the choice problem.
a	Numeric, the parameter of the logit attention rule.

Value

menu	The choice problems.
choice	The simulated choices.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

R. L. Brady and J. Rehbeck (2016). Menu-Dependent Stochastic Feasibility. *Econometrica* 84(3): 1203-1223. doi: 10.3982/ECTA12694

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

Examples

```
set.seed(42)
logitSimu(n = 5, uSize = 6, mSize = 5, a = 2)
```

ramdata *ramdata: Simulated Choice Data*

Description

The file contains a standard choice data of 9,000 observations. There are five alternatives in the grand set.

See `revealPref` for revealed preference analysis, and `revealAtte` for revealed attention. `sumData` is a low-level function that computes summary statistics, and `genMat` generates constraint matrices subject to given preferences.

Format

menu Numeric matrix of 0s and 1s, choice problems (1 indicates an alternative in the choice problem and 0 otherwise).

choice Numeric matrix of 0s and 1s, choices (1 indicates an alternative being chosen).

rAtte *Revealed Preference Analysis in Random Limited Attention Models*

Description

This has been replaced by `revealPref`.

Usage

```
rAtte(
  menu,
  choice,
  pref_list = NULL,
  method = "GMS",
  nCritSimu = 2000,
  BARatio2MS = 0.1,
  BARatio2UB = 0.1,
  MNRatioGMS = NULL,
  RAM = TRUE,
  AOM = TRUE,
  limDataCorr = TRUE,
  attBinary = 1
)
```

Arguments

menu	Numeric matrix of 0s and 1s, the collection of choice problems.
choice	Numeric matrix of 0s and 1s, the collection of choices.
pref_list	Numeric matrix, each row corresponds to one preference. For example, c(2,3,1) means 2 is preferred to 3 and to 1. When set to NULL, the default, c(1,2,3,...), will be used.
method	String, the method for constructing critical values. Default is GMS (generalized moment selection). Other available options are LF (least favorable model), PI (plug-in method), 2MS (two-step moment selection), 2UB (two-step moment upper bound), or ALL (report all critical values).
nCritSimu	Integer, number of simulations used to construct the critical value. Default is 2000.
BARatio2MS	Numeric, beta-to-alpha ratio for two-step moment selection method. Default is 0.1.
BARatio2UB	Numeric, beta-to-alpha ratio for two-step moment upper bound method. Default is 0.1.
MNRatioGMS	Numeric, shrinkage parameter. Default is sqrt(1/log(N)), where N is the sample size.
RAM	Boolean, whether the restrictions implied by the random attention model of Cattaneo, Ma, Masatlioglu, and Suleymanov (2020) should be incorporated, that is, their monotonic attention assumption (default is TRUE).
AOM	Boolean, whether the restrictions implied by the attention overload model of Cattaneo, Cheung, Ma, and Masatlioglu (2021) should be incorporated, that is, their attention overload assumption (default is TRUE).
limDataCorr	Boolean, whether assuming limited data (default is TRUE). When set to FALSE, will assume all choice problems are observed. This option only applies when RAM is set to TRUE.
attBinary	Numeric, between 1/2 and 1 (default is 1), whether additional restrictions (on the attention rule) should be imposed for binary choice problems (i.e., attentive at binaries).

Value

sumStats	Summary statistics, generated by sumData.
constraints	Matrices of constraints, generated by genMat.
Tstat	Test statistic.
critVal	Critical values.
pVal	P-values (only available for GMS, LF and PI).
method	Method for constructing critical value.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

`revealAtte` *Revealed Attention Analysis in Random Limited Attention Models*

Description

Given a random sample of choice problems and choices, `revealAtte` returns the upper and lower bounds on the attention frequency following the construction of Cattaneo, Cheung, Ma, and Masatlioglu (2021).

`sumData` is a low-level function that generates summary statistics. For revealed preference analysis, see `revealPref`.

Usage

```
revealAtte(
  menu,
  choice,
  alternative = NULL,
  S = NULL,
  lower = TRUE,
  upper = TRUE,
  pref = NULL,
  nCritSimu = 2000,
  level = 0.95
)
```

Arguments

`menu`	Numeric matrix of 0s and 1s, the collection of choice problems.
`choice`	Numeric matrix of 0s and 1s, the collection of choices.
`alternative`	Numeric vector, the alternatives for which to compute bounds on the attention frequency. For example, `c(1,2,4)` means the first, second, and fourth alternatives.

S	Numeric matrix of 0s and 1s, the collection of choice problems to compute bounds on the attention frequency.
lower	Boolean, whether lower bounds should be computed (default is TRUE).
upper	Boolean, whether upper bounds should be computed (default is TRUE).
pref	Numeric vector, corresponding to the preference. For example, c(2,3,1) means 2 is preferred to 3 and to 1. When set to NULL, the default, c(1,2,3,...), will be used. This option only applies to the upper bounds (i.e., when upper is set to TRUE).
nCritSimu	Integer, number of simulations used to construct the critical value. Default is 2000.
level	Numeric, the significance level (default is 0.95).

Value

sumStats	Summary statistics, generated by sumData.
lowerBound	Matrix containing the lower bounds.
upperBound	Matrix containing the upper bounds.
critVal	The simulated critical value.
opt	Options used in the function call.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

Examples

```
# Load data
data(ramdata)

# Set seed, to replicate simulated critical values
set.seed(42)

# preference
pref <- matrix(c(1, 2, 3, 4, 5), ncol=5, byrow=TRUE)
# list of choice problems
S <- matrix(c(1, 1, 0, 0, 0,
```

```
              1, 1, 1, 0, 0,
              1, 1, 1, 0, 1,
              1, 1, 1, 1, 1), ncol=5, byrow=TRUE)
result <- revealAtte(menu = ramdata$menu, choice = ramdata$choice,
  alternative = c(1,2), S = S,
  lower = TRUE, upper = TRUE,
  pref = pref)
summary(result)
```

`revealPref` *Revealed Preference Analysis in Random Limited Attention Models*

Description

Given a random sample of choice problems and choices, `revealPref` returns test statistics, critical values and p-values against a collection of preferences. Five methods for choosing critical values are available: (i) `GMS`: generalized moment selection (plug-in (estimated) moment conditions with shrinkage); (ii) `PI`: critical values based on plug-in estimated moment conditions (this is not uniformly valid); (iii) `LF`: critical values based on the least favorable model (plug-in 0 for the moment conditions); (iv) `2MS`: two-step moment selection; and (v) `2UB`: refined moment selection (plug-in upper bound of moment inequalities).

`sumData` is a low-level function that generates summary statistics, and `genMat` can be used to construct the constraint matrices. The simulated dataset `ramdata` is also provided for illustration. For revealed attention analysis, see `revealAtte`.

Usage

```
revealPref(
  menu,
  choice,
  pref_list = NULL,
  method = "GMS",
  nCritSimu = 2000,
  BARatio2MS = 0.1,
  BARatio2UB = 0.1,
  MNRatioGMS = NULL,
  RAM = TRUE,
  AOM = TRUE,
  limDataCorr = TRUE,
  attBinary = 1
)
```

Arguments

`menu`	Numeric matrix of 0s and 1s, the collection of choice problems.
`choice`	Numeric matrix of 0s and 1s, the collection of choices.

pref_list	Numeric matrix, each row corresponds to one preference. For example, c(2,3,1) means 2 is preferred to 3 and to 1. When set to NULL, the default, c(1,2,3,...), will be used.
method	String, the method for constructing critical values. Default is GMS (generalized moment selection). Other available options are LF (least favorable model), PI (plug-in method), 2MS (two-step moment selection), 2UB (two-step moment upper bound), or ALL (report all critical values).
nCritSimu	Integer, number of simulations used to construct the critical value. Default is 2000.
BARatio2MS	Numeric, beta-to-alpha ratio for two-step moment selection method. Default is 0.1.
BARatio2UB	Numeric, beta-to-alpha ratio for two-step moment upper bound method. Default is 0.1.
MNRatioGMS	Numeric, shrinkage parameter. Default is sqrt(1/log(N)), where N is the sample size.
RAM	Boolean, whether the restrictions implied by the random attention model of Cattaneo, Ma, Masatlioglu, and Suleymanov (2020) should be incorporated, that is, their monotonic attention assumption (default is TRUE).
AOM	Boolean, whether the restrictions implied by the attention overload model of Cattaneo, Cheung, Ma, and Masatlioglu (2021) should be incorporated, that is, their attention overload assumption (default is TRUE).
limDataCorr	Boolean, whether assuming limited data (default is TRUE). When set to FALSE, will assume all choice problems are observed. This option only applies when RAM is set to TRUE.
attBinary	Numeric, between 1/2 and 1 (default is 1), whether additional restrictions (on the attention rule) should be imposed for binary choice problems (i.e., attentive at binaries).

Value

sumStats	Summary statistics, generated by sumData.
constraints	Matrices of constraints, generated by genMat.
Tstat	Test statistic.
critVal	Critical values.
pVal	P-values (only available for GMS, LF and PI).
method	Method for constructing critical value.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

Examples

```
# Load data
data(ramdata)

# Set seed, to replicate simulated critical values
set.seed(42)

# list of preferences
pref_list <- matrix(c(1, 2, 3, 4, 5,
                      2, 1, 3, 4, 5,
                      2, 3, 4, 5, 1,
                      5, 4, 3, 2, 1), ncol=5, byrow=TRUE)

# revealed preference using only RAM restrictions
result1 <- revealPref(menu = ramdata$menu, choice = ramdata$choice, method = "GMS",
  pref_list = pref_list, RAM = TRUE, AOM = FALSE)
summary(result1)

# revealed preference using only AOM restrictions
result2 <- revealPref(menu = ramdata$menu, choice = ramdata$choice, method = "GMS",
  pref_list = pref_list, RAM = FALSE, AOM = TRUE)
summary(result2)

# revealed preference using both RAM and AOM restrictions
result3 <- revealPref(menu = ramdata$menu, choice = ramdata$choice, method = "GMS",
  pref_list = pref_list, RAM = TRUE, AOM = TRUE)
summary(result3)

# revealed preference employing additional restrictions for binary choice problems
result4 <- revealPref(menu = ramdata$menu, choice = ramdata$choice, method = "GMS",
  pref_list = pref_list, RAM = TRUE, AOM = TRUE, attBinary = 2/3)
summary(result4)
```

---

sumData *Generate Summary Statistics*

Description

`sumData` generates summary statistics. Given a collection of choice problems and corresponding choices, `sumData` calculates the number of occurrences of each choice problem, as well as the empirical choice probabilities.

This function is embedded in `revealPref`.

Usage

```
sumData(menu, choice)
```

Arguments

menu	Numeric matrix of 0s and 1s, the collection of choice problems.
choice	Numeric matrix of 0s and 1s, the collection of choices.

Value

sumMenu	Summary of choice problems, with repetitions removed.
sumProb	Estimated choice probabilities as sample averages for different choice problems.
sumN	Effective sample size for each choice problem.
sumMsize	Size of each choice problem.
sumProbVec	Estimated choice probabilities as sample averages, collapsed into a column vector.
Sigma	Estimated variance-covariance matrix for the choice rule, scaled by relative sample sizes.

Author(s)

Matias D. Cattaneo, Princeton University. <cattaneo@princeton.edu>.

Paul Cheung, University of Maryland. <hycheung@umd.edu>

Xinwei Ma (maintainer), University of California San Diego. <x1ma@ucsd.edu>

Yusufcan Masatlioglu, University of Maryland. <yusufcan@umd.edu>

Elchin Suleymanov, Purdue University. <esuleyma@purdue.edu>

References

M. D. Cattaneo, P. Cheung, X. Ma, and Y. Masatlioglu (2021). Attention Overload. Working paper.

M. D. Cattaneo, X. Ma, Y. Masatlioglu, and E. Suleymanov (2020). A Random Attention Model. *Journal of Political Economy* 128(7): 2796-2836. doi: 10.1086/706861

Examples

```
# Load data
data(ramdata)

# Generate summary statistics
summaryStats <- sumData(ramdata$menu, ramdata$choice)
nrow(summaryStats$sumMenu)
min(summaryStats$sumN)

summaryStats$sumMenu[1, ]
summaryStats$sumProb[1, ]
summaryStats$sumN[1]
```

Index