Package: ugcflss (via r-universe)

October 15, 2024

Title Unadjusted Group Comparisons For Likert Sum Scores

Version 0.1.2.9001

Description Performs unadjusted group comparisons for data that are
sum scores of several Likert items, and produces plots that
help describe patterns in the data. Models rely on Bayesian
regularization and hierarchical ordinal regression.
Regularization and hierarchical modeling help with stabilizing
model parameters and inference about quantities of interest.
Ordinal regression helps with describing the distribution of
the data accurately.

License GPL (>= 3)

Encoding UTF-8

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.1

Biarch true

Depends R (>= 3.4.0)

Imports bayesplot, data.table, ggplot2, methods, posterior, Rcpp (>=
0.12.0), RcppParallel (>= 5.0.1), rstan (>= 2.18.1), rstantools
(>= 2.4.0), scales, splines2

LinkingTo BH (>= 1.66.0), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.3.0),
RcppParallel (>= 5.0.1), rstan (>= 2.18.1), StanHeaders (>=
2.18.0)

SystemRequirements GNU make

Suggests knitr, rmarkdown, testthat (>= 3.0.0)

Config/testthat/edition 3

VignetteBuilder knitr

Repository https://jamesuanhoro.r-universe.dev

RemoteUrl https://github.com/jamesuanhoro/ugcflss

RemoteRef HEAD

RemoteSha b59fb4c1642d7e2bbed12b04bf02fa21890b3348

Contents

---

ugcflss-package   *The 'ugcflss' package.*

Description

Performs unadjusted group comparisons for data that are sum scores of several Likert items, and produces plots that help describe patterns in the data. Models rely on Bayesian regularization and hierarchical ordinal regression. Regularization and hierarchical modeling help with stabilizing model parameters and inference about quantities of interest. Ordinal regression helps with describing the distribution of the data accurately.

For a demo, check out the vignette, `vignette("ugcflss_tutorial")`

References

Stan Development Team (NA). RStan: the R interface to Stan. R package version 2.32.6. https://mc-stan.org

---

ugcflss_describe   *Compute requested statistic for either entire sample or by group*

Description

Compute requested statistic for either entire sample or by group

Usage

```
ugcflss_describe(
  res_obj,
  by_group = FALSE,
  stat = "median",
  interval = 0.89,
  tau = 0.5,
  return_draws = FALSE
)
```

Arguments

`res_obj`	Object returned by main function
`by_group`	If TRUE, compute by group. If FALSE, compute for entire sample.
`stat`	Statistic of interest, one of "median", "mean", "sd" or "quantile".
`interval`	Some quantile interval between 0 and 1
`tau`	If stat = "quantile", the value of the probability between 0 and 1
`return_draws`	If TRUE, do not summarize the draws. If FALSE, summarize the draws.

Value

Returns dataset.

---

ugcflss_describe_plot   *Plot requested statistic for entire sample and by group*

Description

Plot requested statistic for entire sample and by group

Usage

```
ugcflss_describe_plot(res_obj, stat = "median", interval = 0.89, tau = 0.5)
```

Arguments

`res_obj`	Object returned by main function
`stat`	Statistic of interest, one of "median", "mean", "sd" or "quantile".
`interval`	Some quantile interval between 0 and 1
`tau`	If stat = "quantile", the value of the probability between 0 and 1

Value

Returns plot

---

ugcflss_exceed_all   *Compute CCDF for entire sample*

Description

Compute CCDF for entire sample

Usage

```
ugcflss_exceed_all(res_obj, interval = 0.89, convergence = FALSE)
```

Arguments

res_obj	Object returned by main function
interval	Some quantile interval between 0 and 1
convergence	If TRUE, returns CCDF convergence statistics. If FALSE, creates dataset showing CCDF which can be used for plotting the CCDF.

Value

Returns dataset.

---

ugcflss_exceed_all_plot
                              Plot CCDF for entire sample

Description

Plot CCDF for entire sample

Usage

```
ugcflss_exceed_all_plot(res_obj, interval = 0.89, show_intervals = FALSE)
```

Arguments

res_obj	Object returned by main function
interval	Some quantile interval between 0 and 1
show_intervals	If TRUE, show intervals on plot, if FALSE, do not show intervals

Value

Returns plot

---

ugcflss_exceed_group   *Compute CCDF by group*

Description

Compute CCDF by group

Usage

```
ugcflss_exceed_group(res_obj, interval = 0.89, convergence = FALSE)
```

Arguments

res_obj	Object returned by main function
interval	Some quantile interval between 0 and 1
convergence	If TRUE, returns CCDF convergence statistics. If FALSE, creates dataset showing CCDF which can be used for plotting the CCDF.

Value

Returns dataset.

---

ugcflss_exceed_group_plot

Plot CCDF by group

Description

Plot CCDF by group

Usage

```
ugcflss_exceed_group_plot(res_obj, interval = 0.89, show_intervals = FALSE)
```

Arguments

res_obj	Object returned by main function
interval	Some quantile interval between 0 and 1
show_intervals	If TRUE, show intervals on plot, if FALSE, do not show intervals

Value

Returns plot

---

ugcflss_fit_model   *Analyze a dataset*

Description

Major function for analyzing data. See: `vignette("ugcflss_tutorial")` for a demonstration

Usage

```
ugcflss_fit_model(
  data,
  grouping_variable = NA_character_,
  sum_score = NA_character_,
  minimum_item_response = NA_integer_,
  maximum_item_response = NA_integer_,
  number_items = NA_integer_,
  warmup = 750,
  sampling = 750,
  refresh = max((warmup + sampling)%/%10, 1),
  adapt_delta = 0.9,
  max_treedepth = 10,
  chains = 3,
  cores = min(chains, max(parallel::detectCores() - 2, 1)),
  seed = sample.int(.Machine$integer.max, 1),
  override_twenty_groups = FALSE,
  show_messages = TRUE
)
```

Arguments

data   A dataset, ideally a data.frame.

grouping_variable

                   The name of the grouping_variable in the dataset.

sum_score   The name of the sum_score variable in the dataset. This variable must be whole numbers.

minimum_item_response

                   Theorical minimum value on items summed to create sum score. For example, with Likert data with 5 response categories coded 0,1,2,3,4, this argument would be `minimum_item_response = 0`.

maximum_item_response

                   Theorical maximum value on items summed to create sum score. For example, with Likert data with 3 response categories coded 1,2,3, this argument would be `maximum_item_response = 3`.

number_items   Number of items summed to create the sum score. Must be at least 1.

warmup   Number of iterations used to warmup the sampler, per chain.

sampling	Number of iterations retained for inference, per chain.
refresh	(Positive whole number) How often to print the status of the sampler.
adapt_delta	Number in (0,1). Increase to resolve divergent transitions.
max_treedepth	(Positive whole number) Increase to resolve problems with maximum tree depth.
chains	Number of chains to use.
cores	Number of cores to use.
seed	Random seed.
override_twenty_groups	
	By default, we assume your grouping_variable does not have more than 20 levels. If you want to override this default, add override_twenty_groups = TRUE to your function call.
show_messages	(Logical) If TRUE, show messages from Stan sampler, if FALSE, hide messages.

Value

Object containing analysis results. See: `vignette("ugcflss_tutorial")` for a demonstration.

---

ugcflss_pairwise	*Perform pairwise comparisons of requested statistic*

Description

Perform pairwise comparisons of requested statistic

Usage

```
ugcflss_pairwise(
  res_obj,
  stat = "median",
  interval = 0.89,
  tau = 0.5,
  comparison = "difference",
  return_draws = FALSE
)
```

Arguments

res_obj	Object returned by main function
stat	Statistic of interest, one of "median", "mean", "sd", "quantile" or "ps". ps = probability of superiority.
interval	Some quantile interval between 0 and 1
tau	If stat = "quantile", the value of the probability between 0 and 1
comparison	One of difference, ratio or log-ratio. Log-ratio will lead to an error if any of the values to be logged are negative though this should not happen.
return_draws	If TRUE, do not summarize the draws. If FALSE, summarize the draws.

Value

Returns dataset

---

ugcflss_pairwise_plot   *Perform pairwise comparisons of requested statistic*

Description

Perform pairwise comparisons of requested statistic

Usage

```
ugcflss_pairwise_plot(
  res_obj,
  stat = "median",
  interval = 0.89,
  tau = 0.5,
  comparison = "difference"
)
```

Arguments

res_obj	Object returned by main function
stat	Statistic of interest, one of "median", "mean", "sd", "quantile" or "ps". ps = probability of superiority.
interval	Some quantile interval between 0 and 1
tau	If stat = "quantile", the value of the probability between 0 and 1
comparison	One of difference, ratio or log-ratio. Log-ratio will lead to an error if any of the values to be logged are negative though this should not happen.

Value

Returns dataset

ugcflss_ppd   *Compute posterior predictive distribution*

Description

Compute posterior predictive distribution

Usage

```
ugcflss_ppd(res_obj, ppd_samples = 200)
```

Arguments

res_obj	Object returned by main function
ppd_samples	Number of iterations to use to obtain the posterior predictive distribution.

Value

Returns dataset

ugcflss_ppd_plot   *Basic PPD density overlay plot*

Description

Basic PPD density overlay plot

Usage

```
ugcflss_ppd_plot(res_obj, ppd_samples = 200, by_group = FALSE)
```

Arguments

res_obj	Object returned by main function
ppd_samples	Number of iterations to use to obtain the posterior predictive distribution.
by_group	If TRUE, plot posterior predictive distribution by group. If FALSE, plot for entire sample.

Value

Returns density overlay plot

Index