# Package: DCPO (via r-universe)

October 29, 2026

**Version** 0.5.3

**Title** Dynamic Comparative Public Opinion

**Description** Estimates latent variables of public opinion cross-nationally and over time from sparse and incomparable survey data. 'DCPO' uses a population-level graded response model with country-specific item bias terms. Sampling is conducted with 'Stan'. References: Solt (2020) <doi:10.31235/osf.io/d5n9p>.

**License** GPL (>= 3)

**Encoding** UTF-8

**LazyData** true

**ByteCompile** true

**Depends** R (>= 3.4.0), Rcpp (>= 0.12.17), methods

**Imports** rstan (>= 2.18.1), rstantools (>= 2.0.0), beepr, dplyr, forcats, janitor, purrr, tibble, tidyr

**LinkingTo** StanHeaders (>= 2.18.0), rstan (>= 2.18.1), BH (>= 1.66.0-1), Rcpp (>= 0.12.0), RcppEigen (>= 0.3.3.4.0)

**Suggests** knitr

**SystemRequirements** GNU make

**NeedsCompilation** yes

**RoxygenNote** 7.0.0

**Biarch** true

**Author** Frederick Solt [aut, cre], Trustees of Columbia University [cph]

**Maintainer** Frederick Solt <frederick-solt@uiowa.edu>

**Config/pak/sysreqs** make

**Repository** https://cran.r-universe.dev

**Date/Publication** 2020-05-29 11:50:02 UTC

**RemoteUrl** https://github.com/cran/DCPO

**RemoteRef** HEAD

**RemoteSha** 5fa45405be9006bece967694d380cd75cdbbab1d

# Contents

DCPO-package . . . . . 2
dcpo . . . . . 2
dcpo_xvt . . . . . 3
demsup_data . . . . . 4
get_xvt_results . . . . . 5
summarize_dcpo_results . . . . . 6

**Index** **7**

---

`DCPO-package` *DCPO: Dynamic Comparative Public Opinion*

---

## Description

DCPO estimates dynamic comparative public opinion as a latent variable from survey data

## References

Stan Development Team (2018). RStan: the R interface to Stan. R package version 2.18.2. http://mc-stan.org

---

`dcpo` *Estimate Dynamic Comparative Public Opinion*

---

## Description

`dcpo` uses diverse survey data to estimate public opinion across countries and over time.

## Usage

```
dcpo(dcpo_input, chime = TRUE, ...)
```

## Arguments

| | |
|---|---|
| `dcpo_input` | a data frame of survey items and marginals generated by `DCPOtools::dcpo_setup` |
| `chime` | play chime when complete? |
| `...` | arguments to be passed to `rstan::stan`. Defaults reset by `dcpo` are described below under details. |

## Details

`dcpo`, when passed a data frame `dcpo_input` of survey marginals created by `dcpo_setup`, estimates a latent variable of public opinion. See `rstan::stan` for additional options; `stan` defaults reset by `dcpo` are `seed = 324`, `thin = 2`, `cores = min(stan_args$chains, parallel::detectCores()/2)`, and `control <- list(adapt_delta = 0.99, stepsize = 0.005, max_treedepth = 14)`

## Value

a stanfit object

## Examples

```
out1 <- dcpo(demsup_data,
             chains = 1,
            iter = 300) # 1 chain/300 iterations for example purposes only; use defaults
```

---

dcpo_xvt *Cross-validation testing for DCPO*

---

## Description

dcpo_xvt performs a single cross-validation test for DCPO

## Usage

```
dcpo_xvt(
  dcpo_input,
  fold_number = 1,
  number_of_folds = 10,
  fold_seed = 324,
  chime = TRUE,
  ...
)
```

## Arguments

| | |
|---|---|
| dcpo_input | a data frame of survey items and marginals generated by DCPOtools::dcpo_setup |
| fold_number | an integer indicating the number of the fold to treated as test data in the current analysis |
| number_of_folds | an integer indicating the total number of folds |
| fold_seed | a seed for reproducibly randomly assigning observations to folds; when a complete set of k-fold cross-validations is to be performed, the same seed should be used for all |
| chime | play chime when complete? |
| ... | arguments to be passed to rstan::stan. See dcpo. |

## Details

dcpo_xvt performs a single cross-validation test of a DCPO estimation. To perform a complete k-fold cross-validation, call it repeatedly, changing only the fold_number argument.

## Value

a stanfit object

## Examples

```
# Single cross-validation test with 25% test set
demsup_xvtest_25pct <- dcpo_xvt(demsup_data,
                            number_of_folds = 4,
                            iter = 300,
                        chains = 1) # 1 chain/300 iterations for example only; use defaults
```

---

# demsup_data *Support for Democracy in 51 Survey Datasets*

---

## Description

A dataset containing the prices and other attributes of almost 54,000 diamonds.

## Usage

```
demsup_data
```

## Format

A list of 15 elements

**K** an integer, the total number of countries in the data

**T** an integer, the total number of years in the data

**Q** an integer, the total number of distinct survey questions in the data

**R** an integer, the maximum number of response cutpoints in any survey question in the data

**N** an integer, the number of KTQR observations

**kk** a numeric vector of length N, the country of each observation

**tt** a numeric vector of length N, the year of each observation

**qq** a numeric vector of length N, the question of each observation

**rr** a numeric vector of length N, the response cutpoint of each observation

**y_r** a numeric vector of length N, the number of respondents who provided a response above the relevant cutpoint for each observation

**n_r** a numeric vector of length N, the total number of respondents for each observation

**fixed_cutp** a QxR matrix, a truth table indicating the question-cutpoint to be fixed at difficulty .5

**use_delta** a QxK tibble, a truth table indicating whether item difficulty should be estimated to vary by question-country to account for potential item-response bias

**data** an Nx14 tibble, the aggregate survey response dataset in its original format

**data_args** a list of length 3, indicating the arguments passed to DCPOtools::format_dcpo to generate demsup_data from demsup_data$data

## Details

Data on aggregate support for democracy reported in 51 survey datasets in 998 country-years, formatted for use with the functions of the DCPO package

## Source

demsup_data replicates the data employed in Claassen, Christopher. 2019. "Estimating Smooth Country-Year Panels of Public Opinion." Political Analysis 27(1):1-20. See https://github.com/fsolt/DCPOtools.

---

`get_xvt_results` *Get results of DCPO cross-validation testing*

---

## Description

`get_xvt_results` performs a single cross-validation test for dcpo's estimates of cross-national public opinion

## Usage

```
get_xvt_results(dcpo_xvt_output, ci = 80)
```

## Arguments

| | |
|---|---|
| `dcpo_xvt_output` | output from a single call to `DCPO::dcpo_xvt` or a k-fold test list of such output generated by purrr::map |
| `ci` | an integer indicating the desired width of credible interval for coverage testing; 80 is the default. |

## Value

a stanfit object

## Examples

```
# Single cross-validation test with 25% test set
demsup_xvtest_25pct <- dcpo_xvt(demsup_data,
                            chime = FALSE,
                            number_of_folds = 4,
                            iter = 300,
                          chains = 1) # 1 chain/300 iterations for example only; use defaults

get_xvt_results(demsup_xvtest_25pct)
```

summarize_dcpo_results *Extract DCPO Results*

## Description

summarize_dcpo_results is a convenience function that produces summary statistics of the main parameters of a DCPO stanfit object along with the relevant identifying information (country, year, question, and cutpoint).

## Usage

```
summarize_dcpo_results(
  dcpo_input,
  dcpo_output,
  pars = c("theta", "sigma", "alpha", "beta", "delta"),
  probs = c(0.1, 0.9)
)
```

## Arguments

| | |
|---|---|
| dcpo_input | the data frame of survey items and marginals generated by DCPOtools::dcpo_setup previously passed to DCPO::dcpo to generate the stanfit object passed as dcpo_output |
| dcpo_output | a stanfit object output by DCPO::dcpo |
| pars | a character vector of parameter names to be summarized from the DCPO model: theta (mean public opinion), sigma (polarization in public opinion), alpha (question dispersion), beta (question-cutpoint difficulty), and/or delta (country-specific question bias) |
| probs | a numeric vector of quantiles of interest; the default is c(.1, .9) |

## Value

a tibble

## Examples

```
out1 <- dcpo(demsup_data,
             chime = FALSE,
             chains = 1,
            iter = 300) # 1 chain/300 iterations for example purposes only; use defaults

theta_results <- summarize_dcpo_results(dcpo_input = demsup_data,
                                        dcpo_output = out1,
                                        pars = "theta")
```

# Index

∗ **datasets**
demsup_data, 4

DCPO *(*DCPO-package*)*, 2
dcpo, 2
DCPO-package, 2
dcpo_xvt, 3
demsup_data, 4

get_xvt_results, 5

summarize_dcpo_results, 6