Package ‘statConfR’

October 23, 2024

Type Package

Title Models of Decision Confidence and Measures of Metacognition

Version 0.2.0

Date 2024-14-10

Maintainer Manuel Rausch <manuel.rausch@hochschule-rhein-waal.de>

Description Provides fitting functions and other tools for decision confidence and metacognition researchers, including meta-d'/d', often considered to be the gold standard to measure metacognitive efficiency, and information-theoretic measures of metacognition.
Also allows to fit several static models of decision making and confidence.

License GPL(>=3)

URL https://github.com/ManuelRausch/StatConfR

BugReports https://github.com/ManuelRausch/StatConfR/issues

Depends R (>= 4.0)

Imports parallel, plyr, stats, utils

Date/Publication 2023-09-12

Encoding UTF-8

LazyData true

NeedsCompilation no

Repository CRAN

Roxygen list(old_usage = TRUE, markdown=TRUE)

RoxygenNote 7.3.2

Contents

estimateMetaI *Estimate Measures of Metacognition from Information Theory*

Description

estimateMetaI estimates meta-I, an information-theoretic measure of metacognitive sensitivity proposed by Dayan (2023), as well as similar derived measures, including meta-I_1^r and Meta-I_2^r. These are different normalizations of meta-I:

- Meta-I_1^r normalizes by the meta-I that would be expected from an underlying normal distribution with the same sensitivity.
- Meta-$I_1^{r\prime}$ is a variant of meta-I_1^r not discussed by Dayan (2023) which normalizes by the meta-I that would be expected from an underlying normal distribution with the same accuracy (this is similar to the sensitivity approach but without considering variable thresholds).
- Meta-I_2^r normalizes by the maximum amount of meta-I which would be reached if all uncertainty about the stimulus was removed.
- RMI normalizes meta-I by the range of its possible values and therefore scales between 0 and 1. RMI is a novel measure not discussed by Dayan (2023).

All measures can be calculated with a bias-reduced variant for which the observed frequencies are taken as underlying probability distribution to estimate the sampling bias. The estimated bias is then subtracted from the initial measures. This approach uses Monte-Carlo simulations and is therefore not deterministic (values can vary from one evaluation of the function to the next). However, this is a simple way to reduce the bias inherent in these measures.

Usage

```
estimateMetaI(data, bias_reduction = TRUE)
```

Arguments

`data` a `data.frame` where each row is one trial, containing following variables:

- `participant` (some group ID, most often a participant identifier; the meta-I measures are estimated for each subset of `data` determined by the different values of this column),
- `stimulus` (stimulus category in a binary choice task, should be a factor with two levels, otherwise it will be transformed to a factor with a warning),
- `rating` (discrete confidence judgments, should be a factor with levels ordered from lowest confidence to highest confidence; otherwise will be transformed to factor with a warning),
- `correct` (encoding whether the response was correct; should be 0 for incorrect responses and 1 for correct responses)

`bias_reduction` `logical`. Whether to apply the bias reduction or not. If runtime is too long, consider setting this to FALSE (default: TRUE).

Details

Meta-I is defined as the mutual information between the confidence and accuracy and is calculated as the transmitted information minus the minimal information given the accuracy,

$$meta - I = I(Y; \hat{Y}, C) - I(Y; \hat{Y}).$$

This is equivalent to Dayan's formulation where meta-I is the information that confidence transmits about the correctness of a response,

$$meta - I = I(Y = \hat{Y}; C).$$

Meta-I is expressed in bits, i.e. the log base is 2). The other measures are different normalizations of meta-I and are unitless. It should be noted that Dayan (2023) pointed out that a liberal or conservative use of the confidence levels will affected th mutual information and thus influence meta-I.

Value

a `data.frame` with one row for each subject and the following columns: `participant` is the participant ID, `meta_I` is the estimated meta-I value (expressed in bits, i.e. log base is 2), `meta_Ir1` is meta-I_1^r, `meta_Ir1_acc` is meta-$I_1^{r\prime}$, `meta_Ir2` is meta-I_2^r, and RMI is RMI. with , or unitless for the normalized measures)

Author(s)

Sascha Meyen, <saschameyen@gmail.com>

References

Dayan, P. (2023). Metacognitive Information Theory. Open Mind, 7, 392–411. https://doi.org/10.1162/opmi_a_00091

Examples

```
# 1. Select two subjects from the masked orientation discrimination experiment
data <- subset(MaskOri, participant %in% c(1:2))
head(data)

# 2. Calculate meta-I measures with bias reduction (this may take 10 s per subject)

metaIMeasures <- estimateMetaI(data)


# 3. Calculate meta-I measures for all participants without bias reduction (much faster)
metaIMeasures <- estimateMetaI(MaskOri, bias_reduction = FALSE)
metaIMeasures
```

fitConf *Fit a static confidence model to data*

Description

The `fitConf` function fits the parameters of one static model of decision confidence, provided by the `model` argument, to binary choices and confidence judgments. See Details for the mathematical specification of the implemented models and their parameters. Parameters are fitted using a maximum likelihood estimation method with a initial grid search to find promising starting values for the optimization. In addition, several measures of model fit (negative log-likelihood, BIC, AIC, and AICc) are computed, which can be used for a quantitative model evaluation.

Usage

```
fitConf(data, model = "SDT", nInits = 5, nRestart = 4)
```

Arguments

`data` a `data.frame` where each row is one trial, containing following variables:

- `diffCond` (optional; different levels of discriminability, should be a factor with levels ordered from hardest to easiest),
- `rating` (discrete confidence judgments, should be a factor with levels ordered from lowest confidence to highest confidence; otherwise will be transformed to factor with a warning),
- `stimulus` (stimulus category in a binary choice task, should be a factor with two levels, otherwise it will be transformed to a factor with a warning),
- `correct` (encoding whether the response was correct; should be 0 for incorrect responses and 1 for correct responses)

`model` `character` of length 1. The generative model that should be fitted. Models implemented so far: 'WEV', 'SDT', 'GN', 'PDA', 'IG', 'ITGc', 'ITGcm', 'logN', and 'logWEV'.

`nInits` `integer`. Number of starting values used for maximum likelihood optimization. Defaults to 5.

`nRestart` `integer`. Number of times the optimization algorithm is restarted. Defaults to 4.

Details

The fitting routine first performs a coarse grid search to find promising starting values for the maximum likelihood optimization procedure. Then the best `nInits` parameter sets found by the grid search are used as the initial values for separate runs of the Nelder-Mead algorithm implemented in `optim`. Each run is restarted `nRestart` times.

Mathematical description of models:

The computational models are all based on signal detection theory (Green & Swets, 1966). It is assumed that participants select a binary discrimination response R about a stimulus S. Both S and R can be either -1 or 1. R is considered correct if $S = R$. In addition, we assume that there are K different levels of stimulus discriminability in the experiment, i.e. a physical variable that makes the discrimination task easier or harder. For each level of discriminability, the function fits a different discrimination sensitivity parameter d_k. If there is more than one sensitivity parameter, we assume that the sensitivity parameters are ordered such as $0 < d_1 < ... < d_K$. The models assume that the stimulus generates normally distributed sensory evidence x with mean $S \times d_k/2$ and variance of 1. The sensory evidence x is compared to a decision criterion c to generate a discrimination response R, which is 1, if x exceeds c and -1 else. To generate confidence, it is assumed that the confidence variable y is compared to another set of criteria $\theta_{R,i}, i = 1, ..., L-1$, depending on the discrimination response R to produce a L-step discrete confidence response. The number of thresholds will be inferred from the number of steps in the `rating` column of `data`. Thus, the parameters shared between all models are:

- sensitivity parameters d_1,...,d_K (K: number of difficulty levels)
- decision criterion c
- confidence criterion $\theta_{-1,1}$,$\theta_{-1,2}$, ..., $\theta_{-1,L-1}$, $\theta_{1,1}$, $\theta_{1,2}$,..., $\theta_{1,L-1}$ (L: number of confidence categories available for confidence ratings)

How the confidence variable y is computed varies across the different models. The following models have been implemented so far:

Signal detection rating model (SDT)*:*
According to SDT, the same sample of sensory evidence is used to generate response and confidence, i.e., $y = x$ and the confidence criteria span from the left and right side of the decision criterion c (Green & Swets, 1966).

Gaussian noise model (GN)*:*
According to the model, y is subject to additive noise and assumed to be normally distributed around the decision evidence value x with a standard deviation σ (Maniscalco & Lau, 2016). The parameter σ is a free parameter.

Weighted evidence and visibility model (WEV)*:*
WEV assumes that the observer combines evidence about decision-relevant features of the stimulus with the strength of evidence about choice-irrelevant features to generate confidence (Rausch et al., 2018). Here, we use the version of the WEV model used by Rausch et al. (2023), which assumes that y is normally distributed with a mean of $(1-w) \times x + w \times d_k \times R$ and standard deviation σ. The parameter σ quantifies the amount of unsystematic variability contributing to confidence judgments but not to the discrimination judgments. The parameter w represents the weight that is put on the choice-irrelevant features in the confidence judgment. w and σ are fitted in addition to the set of shared parameters.

Post-decisional accumulation model (PDA)*:*
PDA represents the idea of on-going information accumulation after the discrimination choice (Rausch et al., 2018). The parameter b indicates the amount of additional accumulation. The confidence variable is normally distributed with mean $x + S \times d_k \times b$ and variance b. For this model the parameter b is fitted in addition to the set of shared parameters.

Independent Gaussian model (IG)*:*
According to IG, y is sampled independently from x (Rausch & Zehetleitner, 2017). y is normally distributed with a mean of $a \times d_k$ and variance of 1 (again as it would scale with m). The free parameter m represents the amount of information available for confidence judgment relative to amount of evidence available for the discrimination decision and can be smaller as well as greater than 1.

Independent truncated Gaussian model: HMetad-Version (ITGc)*:*
According to the version of ITG consistent with the HMetad-method (Fleming, 2017; see Rausch et al., 2023), y is sampled independently from x from a truncated Gaussian distribution with a location parameter of $S \times d_k \times m/2$ and a scale parameter of 1. The Gaussian distribution of y is truncated in a way that it is impossible to sample evidence that contradicts the original decision: If $R = -1$, the distribution is truncated to the right of c. If $R = 1$, the distribution is truncated to the left of c. The additional parameter m represents metacognitive

efficiency, i.e., the amount of information available for confidence judgments relative to amount of evidence available for discrimination decisions and can be smaller as well as greater than 1.

Independent truncated Gaussian model: Meta-d'-Version (ITGcm): According to the version of the ITG consistent with the original meta-d' method (Maniscalco & Lau, 2012, 2014; see Rausch et al., 2023), y is sampled independently from x from a truncated Gaussian distribution with a location parameter of $S \times d_k \times m/2$ and a scale parameter of 1. If $R = -1$, the distribution is truncated to the right of $m \times c$. If $R = 1$, the distribution is truncated to the left of $m \times c$. The additional parameter m represents metacognitive efficiency, i.e., the amount of information available for confidence judgments relative to amount of evidence available for the discrimination decision and can be smaller as well as greater than 1.

Logistic noise model (logN): According to logN, the same sample of sensory evidence is used to generate response and confidence, i.e., $y = x$ just as in SDT (Shekhar & Rahnev, 2021). However, according to logN, the confidence criteria are not assumed to be constant, but instead they are affected by noise drawn from a lognormal distribution. In each trial, $\theta_{-1,i}$ is given by $c - \epsilon_i$. Likewise, $\theta_{1,i}$ is given by $c + \epsilon_i$. ϵ_i is drawn from a lognormal distribution with the location parameter $\mu_{R,i} = log(|\overline{\theta}_{R,i} - c|) - 0.5 \times \sigma^2$ and scale parameter σ. σ is a free parameter designed to quantify metacognitive ability. It is assumed that the criterion noise is perfectly correlated across confidence criteria, ensuring that the confidence criteria are always perfectly ordered. Because $\theta_{-1,1}$, ..., $\theta_{-1,L-1}$, $\theta_{1,1}$, ..., $\theta_{1,L-1}$ change from trial to trial, they are not estimated as free parameters. Instead, we estimate the means of the confidence criteria, i.e., $\overline{\theta}_{-1,1}, ..., \overline{\theta}_{-1,L-1}, \overline{\theta}_{1,1}, ...\overline{\theta}_{1,L-1}$, as free parameters.

Logistic weighted evidence and visibility model (logWEV): logWEV is a combination of logN and WEV proposed by Shekhar and Rahnev (2023). Conceptually, logWEV assumes that the observer combines evidence about decision-relevant features of the stimulus with the strength of evidence about choice-irrelevant features (Rausch et al., 2018). The model also assumes that noise affecting the confidence decision variable is lognormal in accordance with Shekhar and Rahnev (2021). According to logWEV, the confidence decision variable y is equal to $y^* \times R$. y^* is sampled from a lognormal distribution with a location parameter of $(1 - w) \times x \times R + w \times d_k$ and a scale parameter of σ. The parameter σ quantifies the amount of unsystematic variability contributing to confidence judgments but not to the discrimination judgments. The parameter w represents the weight that is put on the choice-irrelevant features in the confidence judgment. w and σ are fitted in addition to the set of shared parameters.

Value

Gives data frame with one row and one column for each of the fitted parameters of the selected model as well as additional information about the fit (`negLogLik` (negative log-likelihood of the final set of parameters), `k` (number of parameters), `N` (number of data rows), `AIC` (Akaike Information Criterion; Akaike, 1974), `BIC` (Bayes information criterion; Schwarz, 1978), and `AICc` (AIC corrected for small samples; Burnham & Anderson, 2002))

Author(s)

Sebastian Hellmann, `<sebastian.hellmann@ku.de>`
Manuel Rausch, `<manuel.rausch@ku.de>`

References

Akaike, H. (1974). A New Look at the Statistical Model Identification. IEEE Transactions on Automatic Control, AC-19(6), 716–723.doi: 10.1007/978-1-4612-1694-0_16

Burnham, K. P., & Anderson, D. R. (2002). Model selection and multimodel inference: A practical information-theoretic approach. Springer.

Fleming, S. M. (2017). HMeta-d: Hierarchical Bayesian estimation of metacognitive efficiency from confidence ratings. Neuroscience of Consciousness, 1, 1–14. doi: 10.1093/nc/nix007

Green, D. M., & Swets, J. A. (1966). Signal detection theory and psychophysics. Wiley.

Maniscalco, B., & Lau, H. (2012). A signal detection theoretic method for estimating metacognitive sensitivity from confidence ratings. Consciousness and Cognition, 21(1), 422–430.

Maniscalco, B., & Lau, H. C. (2014). Signal Detection Theory Analysis of Type 1 and Type 2 Data: Meta-d', Response- Specific Meta-d', and the Unequal Variance SDT Model. In S. M. Fleming & C. D. Frith (Eds.), The Cognitive Neuroscience of Metacognition (pp. 25–66). Springer. doi: 10.1007/978-3-642-45190-4_3

Maniscalco, B., & Lau, H. (2016). The signal processing architecture underlying subjective reports of sensory awareness. Neuroscience of Consciousness, 1, 1–17. doi: 10.1093/nc/niw002

Rausch, M., Hellmann, S., & Zehetleitner, M. (2018). Confidence in masked orientation judgments is informed by both evidence and visibility. Attention, Perception, and Psychophysics, 80(1), 134–154. doi: 10.3758/s13414-017-1431-5

Rausch, M., Hellmann, S., & Zehetleitner, M. (2023). Measures of metacognitive efficiency across cognitive models of decision confidence. Psychological Methods. doi: 10.31234/osf.io/kdz34

Rausch, M., & Zehetleitner, M. (2017). Should metacognition be measured by logistic regression? Consciousness and Cognition, 49, 291–312. doi: 10.1016/j.concog.2017.02.007

Schwarz, G. (1978). Estimating the dimension of a model. The Annals of Statistics, 6(2), 461–464. doi: 10.1214/aos/1176344136

Shekhar, M., & Rahnev, D. (2021). The Nature of Metacognitive Inefficiency in Perceptual Decision Making. Psychological Review, 128(1), 45–70. doi: 10.1037/rev0000249

Shekhar, M., & Rahnev, D. (2023). How Do Humans Give Confidence? A Comprehensive Comparison of Process Models of Perceptual Metacognition. Journal of Experimental Psychology: General. doi:10.1037/xge0001524

Examples

```
# 1. Select one subject from the masked orientation discrimination experiment
data <- subset(MaskOri, participant == 1)
head(data)

# 2. Use fitting function

  # Fitting takes some time (about 10 minutes on an 2.8GHz processor) to run:
  FitFirstSbjWEV <- fitConf(data, model="WEV")
```

fitConfModels *Fit several static confidence models to multiple participants*

Description

The `fitConfModels` function fits the parameters of several computational models of decision confidence, in binary choice tasks, specified in the `model` argument, to different subsets of one data frame, indicated by different values in the column `participant` of the `data` argument. `fitConfModels` is a wrapper of the function `fitConf` and calls `fitConf` for every possible combination of model in the `models` argument and sub-data frame of `data` for each value in the `participant` column. See Details for more information about the parameters. Parameters are fitted using a maximum likelihood estimation method with a initial grid search to find promising starting values for the optimization. In addition, several measures of model fit (negative log-likelihood, BIC, AIC, and AICc) are computed, which can be used for a quantitative model evaluation.

Usage

```
fitConfModels(data, models = "all", nInits = 5, nRestart = 4,
  .parallel = FALSE, n.cores = NULL)
```

Arguments

`data` a `data.frame` where each row is one trial, containing following variables:

- `diffCond` (optional; different levels of discriminability, should be a factor with levels ordered from hardest to easiest),
- `rating` (discrete confidence judgments, should be a factor with levels ordered from lowest confidence to highest confidence; otherwise will be transformed to factor with a warning),
- `stimulus` (stimulus category in a binary choice task, should be a factor with two levels, otherwise it will be transformed to a factor with a warning),
- `correct` (encoding whether the response was correct; should be 0 for incorrect responses and 1 for correct responses)
- `participant` (some group ID, most often a participant identifier; the models given in the second argument are fitted to each subset of `data` determined by the different values of this column)

`models` `character`. The different computational models that should be fitted. Models implemented so far: 'WEV', 'SDT', 'GN', 'PDA', 'IG', 'ITGc', 'ITGcm', 'logN', and 'logWEV'. Alternatively, if `model="all"` (default), all implemented models will be fit.

`nInits` `integer`. Number of initial values used for maximum likelihood optimization. Defaults to 5.

`nRestart` `integer`. Number of times the optimization is restarted. Defaults to 4.

`.parallel` `logical`. Whether to parallelize the fitting over models and participant (default: FALSE)

`n.cores` `integer`. Number of cores used for parallelization. If NULL (default), the available number of cores -1 will be used.

Details

The provided `data` argument is split into subsets according to the values of the `participant` column. Then for each subset and each model in the `models` argument, the parameters of the respective model are fitted to the data subset.

The fitting routine first performs a coarse grid search to find promising starting values for the maximum likelihood optimization procedure. Then the best `nInits` parameter sets found by the grid search are used as the initial values for separate runs of the Nelder-Mead algorithm implemented in `optim`. Each run is restarted `nRestart` times.

Mathematical description of models:

The computational models are all based on signal detection theory (Green & Swets, 1966). It is assumed that participants select a binary discrimination response R about a stimulus S. Both S and R can be either -1 or 1. R is considered correct if $S = R$. In addition, we assume that there are K different levels of stimulus discriminability in the experiment, i.e. a physical variable that makes the discrimination task easier or harder. For each level of discriminability, the function fits a different discrimination sensitivity parameter d_k. If there is more than one sensitivity parameter, we assume that the sensitivity parameters are ordered such as $0 < d_1 < d_2 < ... < d_K$. The models assume that the stimulus generates normally distributed sensory evidence x with mean $S \times d_k/2$ and variance of 1. The sensory evidence x is compared to a decision criterion c to generate a discrimination response R, which is 1, if x exceeds c and -1 else. To generate confidence, it is assumed that the confidence variable y is compared to another set of criteria $\theta_{R,i}, i = 1, 2, ..., L-1$, depending on the discrimination response R to produce a L-step discrete confidence response. The number of thresholds will be inferred from the number of steps in the `rating` column of `data`. Thus, the parameters shared between all models are:

- sensitivity parameters d_1,...,d_K (K: number of difficulty levels)
- decision criterion c
- confidence criterion $\theta_{-1,1}$,$\theta_{-1,2}$, ..., $\theta_{-1,L-1}$, $\theta_{1,1}$, $\theta_{1,2}$,..., $\theta_{1,L-1}$ (L: number of confidence categories available for confidence ratings)

How the confidence variable y is computed varies across the different models. The following models have been implemented so far:

Signal detection rating model (SDT)*:*

According to SDT, the same sample of sensory evidence is used to generate response and confidence, i.e., $y = x$ and the confidence criteria span from the left and right side of the decision criterion c(Green & Swets, 1966).

Gaussian noise model (GN)*:*

According to the model, y is subject to additive noise and assumed to be normally distributed around the decision evidence value x with a standard deviation σ(Maniscalco & Lau, 2016). σ is an additional free parameter.

Weighted evidence and visibility model (WEV)*:*

WEV assumes that the observer combines evidence about decision-relevant features of the stimulus with the strength of evidence about choice-irrelevant features to generate confidence (Rausch et al., 2018). Thus, the WEV model assumes that y is normally distributed with a mean of $(1-w) \times x + w \times d_k \times R$ and standard deviation σ. The standard deviation quantifies the amount of unsystematic variability contributing to confidence judgments but not to the discrimination judgments. The parameter w represents the weight that is put on the choice-irrelevant features in the confidence judgment. w and σ are fitted in addition to the set of shared parameters.

Post-decisional accumulation model (PDA)*:*

PDA represents the idea of on-going information accumulation after the discrimination choice (Rausch et al., 2018). The parameter a indicates the amount of additional accumulation. The confidence variable is normally distributed with mean $x + S \times d_k \times a$ and variance a. For this model the parameter a is fitted in addition to the shared parameters.

Independent Gaussian model (IG):
According to IG, y is sampled independently from x (Rausch & Zehetleitner, 2017). y is normally distributed with a mean of $a \times d_k$ and variance of 1 (again as it would scale with m). The additional parameter m represents the amount of information available for confidence judgment relative to amount of evidence available for the discrimination decision and can be smaller as well as greater than 1.

Independent truncated Gaussian model: HMetad-Version (ITGc):
According to the version of ITG consistent with the HMetad-method (Fleming, 2017; see Rausch et al., 2023), y is sampled independently from x from a truncated Gaussian distribution with a location parameter of $S \times d_k \times m/2$ and a scale parameter of 1. The Gaussian distribution of y is truncated in a way that it is impossible to sample evidence that contradicts the original decision: If $R = -1$, the distribution is truncated to the right of c. If $R = 1$, the distribution is truncated to the left of c. The additional parameter m represents metacognitive efficiency, i.e., the amount of information available for confidence judgments relative to amount of evidence available for discrimination decisions and can be smaller as well as greater than 1.

Independent truncated Gaussian model: Meta-d'-Version (ITGcm):
According to the version of the ITG consistent with the original meta-d' method (Maniscalco & Lau, 2012, 2014; see Rausch et al., 2023), y is sampled independently from x from a truncated Gaussian distribution with a location parameter of $S \times d_k \times m/2$ and a scale parameter of 1. If $R = -1$, the distribution is truncated to the right of $m \times c$. If $R = 1$, the distribution is truncated to the left of $m \times c$. The additional parameter m represents metacognitive efficiency, i.e., the amount of information available for confidence judgments relative to amount of evidence available for the discrimination decision and can be smaller as well as greater than 1.

Logistic noise model (logN):
According to logN, the same sample of sensory evidence is used to generate response and confidence, i.e., $y = x$ just as in SDT (Shekhar & Rahnev, 2021). However, according to logN, the confidence criteria are not assumed to be constant, but instead they are affected by noise drawn from a lognormal distribution. In each trial, $\theta_{-1,i}$ is given by $c - \epsilon_i$. Likewise, $\theta_{1,i}$ is given by $c + \epsilon_i$. ϵ_i is drawn from a lognormal distribution with the location parameter $\mu_{R,i} = log(|\overline{\theta}_{R,i} - c|) - 0.5 \times \sigma^2$ and scale parameter σ. σ is a free parameter designed to quantify metacognitive ability. It is assumed that the criterion noise is perfectly correlated across confidence criteria, ensuring that the confidence criteria are always perfectly ordered. Because $\theta_{-1,1}$, ..., $\theta_{-1,L-1}$, $\theta_{1,1}$, ..., $\theta_{1,L-1}$ change from trial to trial, they are not estimated as free parameters. Instead, we estimate the means of the confidence criteria, i.e., $\overline{\theta}_{-1,1}, ..., \overline{\theta}_{-1,L-1}, \overline{\theta}_{1,1}, ...\overline{\theta}_{1,L-1}$, as free parameters.

Logistic weighted evidence and visibility model (logWEV):
logWEV is a combination of logN and WEV proposed by Shekhar and Rahnev (2023). Conceptually, logWEV assumes that the observer combines evidence about decision-relevant features of the stimulus with the strength of evidence about choice-irrelevant features (Rausch et al., 2018). The model also assumes that noise affecting the confidence decision variable is lognormal in accordance with Shekhar and Rahnev (2021). According to logWEV, the confidence decision variable is y is equal to $y^* \times R$. y^* is sampled from a lognormal distribution with a location parameter of $(1 - w) \times x \times R + w \times d_k$ and a scale parameter of σ. The parameter σ quantifies the amount of unsystematic variability contributing to confidence judgments but not to the discrimination judgments. The parameter w represents the weight that is put on the choice-irrelevant features in the confidence judgment. w and σ are fitted in addition to the set of shared parameters.

Value

Gives data frame with one row for each combination of model and participant. Columns include a model and participant column, one column for each estimated parameter for the different models (parameters that are not present in a specific model (row) but in other models are filled with NAs. Additional information about the fit is provided in additional columns:

- `negLogLik` (negative log-likelihood of the best-fitting set of parameters),
- `k` (number of parameters),
- `N` (number of trials),
- `AIC` (Akaike Information Criterion; Akaike, 1974),
- `BIC` (Bayes information criterion; Schwarz, 1978),
- `AICc` (AIC corrected for small samples; Burnham & Anderson, 2002) If length(models) > 1 or models == "all", there will be three additional columns:
- `wAIC`: Akaike weights based on AIC,
- `wAIC`: Akaike weights based on AICc,
- `wBICc`: Schwarz weights (see Burnham & Anderson, 2002)

Author(s)

Sebastian Hellmann, `<sebastian.hellmann@ku.de>`
Manuel Rausch, `<manuel.rausch@ku.de>`

References

Akaike, H. (1974). A New Look at the Statistical Model Identification. IEEE Transactions on Automatic Control, AC-19(6), 716–723.doi: 10.1007/978-1-4612-1694-0_16

Burnham, K. P., & Anderson, D. R. (2002). Model selection and multimodel inference: A practical information-theoretic approach. Springer.

Fleming, S. M. (2017). HMeta-d: Hierarchical Bayesian estimation of metacognitive efficiency from confidence ratings. Neuroscience of Consciousness, 1, 1–14. doi: 10.1093/nc/nix007

Green, D. M., & Swets, J. A. (1966). Signal detection theory and psychophysics. Wiley.

Maniscalco, B., & Lau, H. (2012). A signal detection theoretic method for estimating metacognitive sensitivity from confidence ratings. Consciousness and Cognition, 21(1), 422–430.

Maniscalco, B., & Lau, H. C. (2014). Signal Detection Theory Analysis of Type 1 and Type 2 Data: Meta-d', Response- Specific Meta-d', and the Unequal Variance SDT Model. In S. M. Fleming & C. D. Frith (Eds.), The Cognitive Neuroscience of Metacognition (pp. 25–66). Springer. doi: 10.1007/978-3-642-45190-4_3

Maniscalco, B., & Lau, H. (2016). The signal processing architecture underlying subjective reports of sensory awareness. Neuroscience of Consciousness, 1, 1–17. doi: 10.1093/nc/niw002

Rausch, M., Hellmann, S., & Zehetleitner, M. (2018). Confidence in masked orientation judgments is informed by both evidence and visibility. Attention, Perception, and Psychophysics, 80(1), 134–154. doi: 10.3758/s13414-017-1431-5

Rausch, M., Hellmann, S., & Zehetleitner, M. (2023). Measures of metacognitive efficiency across cognitive models of decision confidence. Psychological Methods. doi: 10.31234/osf.io/kdz34

Rausch, M., & Zehetleitner, M. (2017). Should metacognition be measured by logistic regression? Consciousness and Cognition, 49, 291–312. doi: 10.1016/j.concog.2017.02.007

Schwarz, G. (1978). Estimating the dimension of a model. The Annals of Statistics, 6(2), 461–464. doi: 10.1214/aos/1176344136

Shekhar, M., & Rahnev, D. (2021). The Nature of Metacognitive Inefficiency in Perceptual Decision Making. Psychological Review, 128(1), 45–70. doi: 10.1037/rev0000249

Shekhar, M., & Rahnev, D. (2023). How Do Humans Give Confidence? A Comprehensive Comparison of Process Models of Perceptual Metacognition. Journal of Experimental Psychology: General. doi:10.1037/xge0001524

Examples

```
# 1. Select two subjects from the masked orientation discrimination experiment
data <- subset(MaskOri, participant %in% c(1:2))
head(data)

# 2. Fit some models to each subject of the masked orientation discrimination experiment

  # Fitting several models to several subjects takes quite some time
  # (about 10 minutes per model fit per participant on a 2.8GHz processor
  # with the default values of nInits and nRestart).
  # If you want to fit more than just two subjects,
  # we strongly recommend setting .parallel=TRUE
  Fits <- fitConfModels(data, models = c("SDT", "ITGc"), .parallel = FALSE)
```

`fitMetaDprime` *title Compute measures of metacognitive sensitivity (meta-d') and metacognitive efficiency(meta-d'/d') for data from one or several subjects*

Description

This function computes the measures for metacognitive sensitivity, meta-d', and metacognitive efficiency, meta-d'/d' (Maniscalco and Lau, 2012, 2014; Fleming, 2017) to data from binary choice tasks with discrete confidence judgments. Meta-d' and meta-d'/d' are computed using a maximum likelihood method for each subset of the `data` argument indicated by different values in the column `participant`, which can represent different subjects as well as experimental conditions.

Usage

```
fitMetaDprime(data, model = "ML", nInits = 5, nRestart = 3,
  .parallel = FALSE, n.cores = NULL)
```

Arguments

`data`	a `data.frame` where each row is one trial, containing following variables: • `rating` (discrete confidence judgments, should be given as factor; otherwise will be transformed to factor with a warning), • `stimulus` (stimulus category in a binary choice task, should be a factor with two levels, otherwise it will be transformed to a factor with a warning), • `correct` (encoding whether the response was correct; should be 0 for incorrect responses and 1 for correct responses) • `participant` (giving the subject ID; the models given in the second argument are fitted for each subject individually.
`model`	`character` of length 1. Either "ML" to use the original model specification by Maniscalco and Lau (2012, 2014) or "F" to use the model specification by Fleming (2017)'s HmetaD method. Defaults to "ML"
`nInits`	`integer`. Number of initial values used for maximum likelihood optimization. Defaults to 5.
`nRestart`	`integer`. Number of times the optimization is restarted. Defaults to 3.
`.parallel`	`logical`. Whether to parallelize the fitting over models and participant (default: FALSE)
`n.cores`	`integer`. Number of cores used for parallelization. If NULL (default), the available number of cores -1 will be used.

Details

The function computes meta-d' and meta-d'/d' either using the hypothetical signal detection model assumed by Maniscalco and Lau (2012, 2014) or the one assumed by Fleming (2014).

The conceptual idea of meta-d' is to quantify metacognition in terms of sensitivity in a hypothetical signal detection rating model describing the primary task, under the assumption that participants had perfect access to the sensory evidence and were perfectly consistent in placing their confidence criteria (Maniscalco & Lau, 2012, 2014). Using a signal detection model describing the primary task to quantify metacognition allows a direct comparison between metacognitive accuracy and discrimination performance because both are measured on the same scale. Meta-d' can be compared against the estimate of the distance between the two stimulus distributions estimated from discrimination responses, which is referred to as d': If meta-d' equals d', it means that metacognitive accuracy is exactly as good as expected from discrimination performance. Ifmeta-d' is lower than d', it means that metacognitive accuracy is suboptimal. It can be shown that the implicit model of confidence underlying the meta-d'/d' method is identical to the independent truncated Gaussian model.

The provided `data` argument is split into subsets according to the values of the `participant` column. Then for each subset, the parameters of the hypothetical signal detection model determined by the `model` argument are fitted to the data subset.

The fitting routine first performs a coarse grid search to find promising starting values for the maximum likelihood optimization procedure. Then the best `nInits` parameter sets found by the grid search are used as the initial values for separate runs of the Nelder-Mead algorithm implemented in `optim`. Each run is restarted `nRestart` times. Warning: meta-d'/d' is only guaranteed to be unbiased from discrimination sensitivity, discrimination bias, and confidence criteria if the data is generated according to the independent truncated Gaussian model (see Rausch et al., 2023).

Value

Gives data frame with one row for each participant and following columns:

- `model` gives the model used for the computation of meta-d’ (see `model` argument)
- `participant` is the participant ID for the respecitve row
- `dprime` is the discrimination sensitivity index d, calculated using a standard SDT formula
- `c` is the discrimination bias c, calculated using a standard SDT formula
- `metaD` is meta-d’, discrimination sensitivity estimated from confidence judgments conditioned on the response
- `Ratio` is meta-d’/d’, a quantity usually referred to as metacognitive efficiency.

Author(s)

Manuel Rausch, <manuel.rausch@hochschule-rhein-waal.de>

References

Fleming, S. M. (2017). HMeta-d: Hierarchical Bayesian estimation of metacognitive efficiency from confidence ratings. Neuroscience of Consciousness, 1, 1–14. doi: 10.1093/nc/nix007

Maniscalco, B., & Lau, H. (2012). A signal detection theoretic method for estimating metacognitive sensitivity from confidence ratings. Consciousness and Cognition, 21(1), 422–430.

Maniscalco, B., & Lau, H. C. (2014). Signal Detection Theory Analysis of Type 1 and Type 2 Data: Meta-d’, Response- Specific Meta-d’, and the Unequal Variance SDT Model. In S. M. Fleming & C. D. Frith (Eds.), The Cognitive Neuroscience of Metacognition (pp. 25–66). Springer. doi: 10.1007/978-3-642-45190-4_3

Rausch, M., Hellmann, S., & Zehetleitner, M. (2023). Measures of metacognitive efficiency across cognitive models of decision confidence. Psychological Methods. doi: 10.31234/osf.io/kdz34

Examples

```
# 1. Select two subject from the masked orientation discrimination experiment
data <- subset(MaskOri, participant %in% c(1:2))
head(data)

# 2. Fit meta-d/d for each subject in data
MetaDs <- fitMetaDprime(data, model="F", .parallel = FALSE)
```

MaskOri — *Data of 16 participants in a masked orientation discrimination experiment (Hellmann et al., 2023, Exp. 1)*

Description

In each trial, participants were shown a sinusoidal grating oriented either horizontally or vertically, followed by a mask after varying stimulus-onset-asynchronies. Participants were instructed to report the orientation and their degree of confidence as accurately as possible

Usage

```
data(MaskOri)
```

Format

A data.frame with 25920 rows representing different trials and 5 variables:

participant integer values as unique participant identifier

stimulus orientation of the grating (90: vertical, 0: horizontal)

response participants' orientation judgment about the grating (90: vertical, 0: horizontal)

correct 0-1 column indicating whether the discrimination response was correct (1) or not (0)

rating 0-4 confidence rating on a continous scale binned into five categories

diffCond stimulus-onset-asynchrony in ms (i.e. time between stimulus and mask onset)

trialNo Enumeration of trials per participant

Examples

```
data(MaskOri)
summary(MaskOri)
```

`simConf` *Simulate data according to a static model of confidence*

Description

This function generates a data frame with random trials generated according to the computational model of decision confidence specified in the `model` argument with given parameters. Simulations can be used to visualize and test qualitative model predictions (e.g. using previously fitted parameters returned by `fitConf`). See `fitConf` for a full mathematical description of all models and their parameters.

Usage

```
simConf(model = "SDT", paramDf)
```

Arguments

`model` `character` of length 1. The generative model that should be used for simulation. Models implemented so far: 'WEV', 'SDT', 'GN', 'PDA', 'IG', 'ITGc', 'ITGcm', 'logN', and 'logWEV'.

`paramDf` a `data.frame`providing the number of generared trials and the parameters of the chosen model. `paramDf` should contain following columns (which parameters are needed depends on the specific model):

- `N` (the number of trials be simulated),
- `participant` (optional, the participant ID of each parameter set. Should be unique to each row),
- `d_1`, `d_2`, ... (sensitivity parameters. The number of sensitivity parameters determines the number of levels of discriminability),
- `c` (discrimination bias),
- `theta_minus.1`, `theta_minus.2`, ... (confidence criteria associated with the response R = -1. The function simulates one more confidence category than there are confidence criteria),

- theta_plus.1, theta_plus.2, ... (confidence criteria associated with the response R = 1. The function simulates one more confidence category than there are confidence criteria),
- w (only for models WEV and logWEV: the visibility weighting parameter, bounded between 0 and 1),
- sigma (only for models WEV, GN, logN, and logWEV: confidence noise, bounded between 0 and Inf),
- m (only for IG, ITGm, and ITGcm: metacognitive efficiency parameter, bounded between 0 and Inf),
- b (only for PDA: postdecisional accumulation parameter, bounded between 0 and Inf),
- M_theta_minus.1, M_theta_minus.2, ... (only for logN: Mean confidence criteria associated with the response R = -1),
- M_theta_plus.1, M_theta_plus.2,... (only for logN: Mean confidence criteria associated with the response R = 1).

Details

The function generates about N trials per row with the provided parameters in the data frame. The output includes a column participant indicating the row ID of the simulated data. The values of the participant column may be controlled by the user, by including a participant column in the input paramDf. Note that the values of this column have to be unique! If no participant column is present in the input, the row numbers will be used as row IDs.

The number of simulated trials for each row of parameters may slightly deviate from the provided N. Precisely, if there are K levels of sensitivity (i.e. there are columns d1, d2, ..., dK), the function simulates round(N/2/K) trials per stimulus identity (2 levels) and level of sensitivity (K levels).

Simulation is performed following the generative process structure of the models. See fitConf for a detailed description of the different models.

Value

a dataframe with about nrow(paramDf)*N rows (see Details), and the following columns:

- participant giving the row ID of the simulation (see Details)
- stimulus giving the category of the stimulus (-1 or 1)
- only, if more than 1 sensitivity parameter (d1,d2,...) is provided: diffCond representing the difficulty condition (values correspond to the levels of the sensitivity parameters, i.e. diffCond=1 represents simulated trials with sensitivity d1)
- response giving the response category (-1 or 1, corresponding to the stimulus categories)
- rating giving the discrete confidence rating (integer, number of categories depends on the number of confidence criteria provided in the parameters)
- correct giving the accuracy of the response (0 incorrect, 1 correct)
- ratings same as rating but as a factor

Author(s)

Manuel Rausch, <manuel.rausch@hochschule-rhein-waal.de>

Examples

```
# 1. define some parameters
paramDf <- data.frame(d_1 = 0, d_2 = 2, d_3 = 4,c = .0,
theta_minus.2 = -2, theta_minus.1 = -1, theta_plus.1 = 1, theta_plus.2 = 2,
sigma = 1/2, w = 0.5, N = 500)
# 2. Simulate dataset
SimulatedData <- simConf(model = "WEV", paramDf)
```

Index