The bhsdtr package: a general purpose method of Bayesian inference for Signal Detection Theory models

Borysław Paulewicz

University of Social Sciences and Humanities, Faculty in Katowice, Poland

Agata Blaut

Department of Psychology, Jagiellonian University, Krakow, Poland

Author Note

Correspondence concerning this article should be addressed to Borysław Paulewicz, Department of Psychology, University of Social Sciences and Humanities, Faculty in Katowice, ul. Techników 9, 40-326 Katowice, Poland.

Contact: bpaulewicz@swps.edu.pl

Abstract

We describe a novel method of Bayesian inference for hierarchical or non-hierarchical equal-variance normal Signal Detection Theory models with one or more criteria. The method is implemented as an open-source R package that uses the state-of-the-art platform, Stan, for sampling from posterior distributions. Our method can accommodate binary responses as well as additional ratings and an arbitrary number of nested or crossed random grouping factors. The SDT parameters can be regressed on additional predictors within the same model via intermediate unconstrained parameters, and the model can be extended by using automatically generated human-readable Stan code as a template. In the paper we explain how our method improves on other similar available methods, we give an overview of the package, demonstrate its ease of use by providing a real-study data analysis walk-through, and show that the model successfully recovers known parameter values when fitted to simulated data.

Keywords: Signal Detection Theory, rating experiments, Bayesian inference, hierarchical models

The bhsdtr package: a general purpose method of Bayesian inference for Signal Detection Theory models

Many tasks used in psychology studies are essentially classification tasks. For example, in a memory study participants may be required to decide if a given test item is old or new, or, in a perceptual study, an object may be either a letter or a digit. If a task requires classification, it is always possible that conclusions based on accuracy or percent correct are invalid because the ability to discriminate between stimulus classes (i.e., sensitivity) is confounded with bias, which is a tendency to classify stimuli as belonging to a particular class (Green & Swets, 1966). In principle, any effect that manifests itself in differences in classification accuracy may reflect differences in sensitivity, bias, or both. Signal Detection Theory provides a simple and popular solution to this common and important problem.

However, because the SDT model is non-linear, variability in its parameters due to factors which are usually study-specific and of no direct interest to the researcher such as participants or items has to be accounted for, otherwise the estimates of SDT parameters are biased. As we explain later in this paper, none of the available methods of inference for hierarchical SDT models that we are aware of addresses this problem correctly in it's generality. Hence, our main goal was to create a correct implementation of the general hierarchical linear regression structure defined on SDT parameters. We argue that the `bhsdtr` package for R (R Core Team, 2017) provides exactly such an implementation and we have made it publicly available at `https://github.com/boryspaulewicz/bhsdtr`, together with the annotated source code that was used to perform all the analyses and produce all the figures presented in this paper.

In what follows, after introducing the most common version of the SDT model, we describe its generalization, which can accommodate data from rating experiments. Next, we explain briefly why, if a method of inference for SDT models were to be of general use in psychology studies, it is essential that it is based on a model equipped with the hierarchical linear regression structure. The `bhsdtr` package meets this requirement thanks to a novel parametrization. We describe this novel parametrization and explain how reliance on some popular parametrizations leads to problems in the two other available implementations. We end the first part of this paper with a formal definition of the model as implemented in `bhsdtr`. The second part contains an overview of the package and a tutorial in which we demonstrate how to use our method in practice. Before we go any further, however, a note on terminology seems in order.

In the context of hierarchical modelling, factors such as participants, items, or replications are often referred to as groups. In our opinion this naming convention may be confusing; a single participant is both a group and a member of some group, while at the same time the term "group" is perhaps most strongly associated with study conditions, as in "experimental group". In this paper we use the term "sampled factor" instead because, by virtue of being a new term, it is unambiguous and seems descriptively correct: the term "sampled factor" seems to capture essential properties of such variables, i.e., a nominal scale, the fact that values are sampled from a larger population and are usually not of direct interest, as in "this is only a sample", and that conclusions of statistical analysis are meant to apply to the whole population of possible values.

Equal-variance normal SDT model with additional criteria

According to SDT, each stimulus in a classification task gives rise, by some unspecified cognitive process, to a unidimensional internal evidence value s sampled from a distribution that depends on the stimulus class. For historical reasons the two stimulus classes are often referred to as "noise" and "signal", and task performance is described in terms of hits, correct rejections, omissions, and false alarms, but this terminology is appropriate only when the model is applied to tasks that require detection, which is far from always being the case. In the most widely used version of the model, shown in Fig. 1, the two evidence distributions are normal with the same variance, which is usually fixed at unity to make the model identifiable. The distance d' between the means of the evidence distributions represents sensitivity. Because normal distributions are unbounded, s is always ambiguous, and so a criterion c placed on the evidence axis has to be used to reach a binary decision. A participant is assumed to decide that a stimulus belongs to the first class (e.g., an old item) if $s < c$, or that it belongs to the second class (e.g., a new item) if $s \geq c$. The location of the decision criterion represents bias.

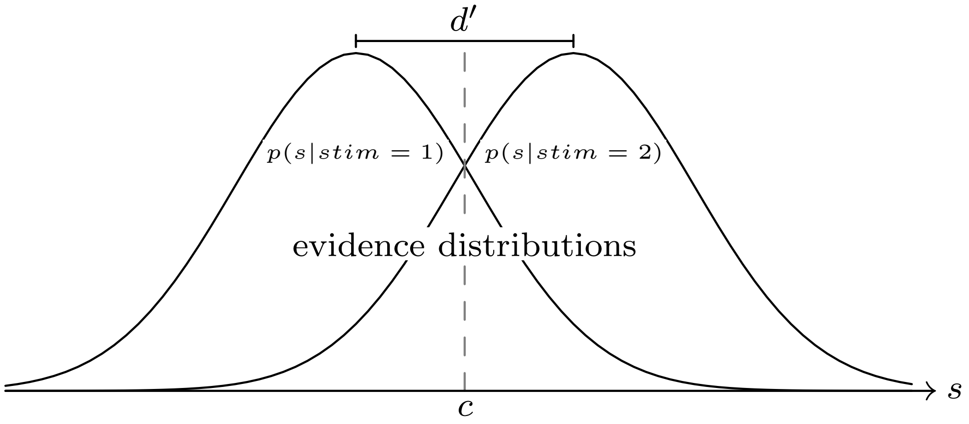

Figure 1. Equal-variance normal Signal Detection Theory model

Perhaps the simplest way of using this model is to fit it to observed response counts and use the estimated d' values in place of the percent correct scores; if the model is correct, the resulting performance measure is not contaminated by bias. Obviously, the model may not be correct, which is one of the reasons why we focus more on the generalized version shown in Fig. 2 below.

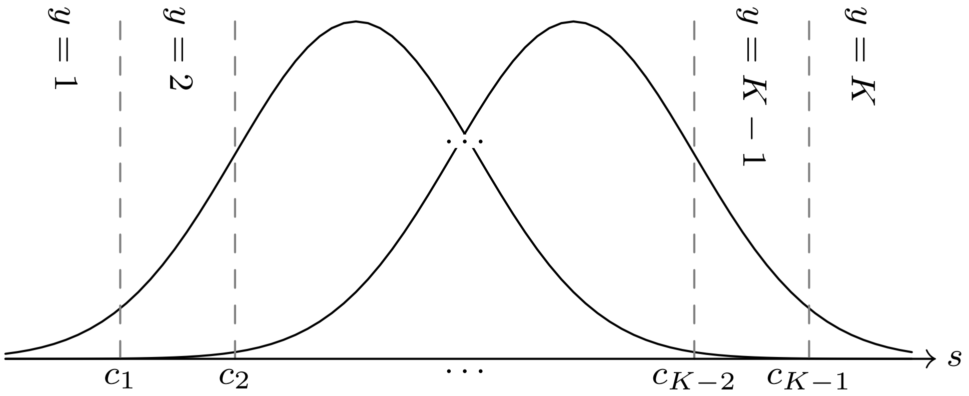

Figure 2. Equal-variance normal Signal Detection Theory model with additional criteria

This generalized model is applicable to studies in which participants are asked to rate their binary classification decisions on confidence or some other performance- or stimulus-related dimension. The ratings and the binary classification decisions can be provided together (e.g., "I am almost certain that it was a digit"), or in an arbitrary order.

Ratings are accommodated by introducing additional criteria and modelling a combined response y, which represents both the binary classification decision and rating as a single number. The value of y increases with the strength of evidence in favor of the second stimulus class. For example, if confidence is rated on a four-point scale, then $y = 1$ when a subject decides that a stimulus belongs to the first class with certainty 4, and $y = 5$ when a subject decides that a stimulus belongs to the second class with certainty 1. More formally, if K is the number of possible combined responses, then a subject is assumed to give response k if $s \in (\boldsymbol{c}_{k-1}, \boldsymbol{c}_k]$, where $\boldsymbol{c}$ are the decision criteria, with $\boldsymbol{c}_0$ and $\boldsymbol{c}_K$ fixed at $-\infty$ and $+\infty$ respectively.

There is a good reason to collect the ratings and use the generalized SDT model from Fig. 2, even when neither the ratings nor the placement of criteria are relevant to the research problem. When $K = 2$ (no ratings), the SDT model fits perfectly, regardless of whether it is a reasonably good approximation to reality, because the data and the model have the same dimensionality. This makes the generalization to the $K > 2$ case particularly important, as it is only when $K > 2$ that the formal assumptions of the model (e.g., equal or unequal variance) can be tested[1], which is often done by comparing observed and predicted ROC curves.

When SDT models are used in psychology studies, researchers are usually interested not in the values of the SDT parameters themselves, but in relationships between these parameters and additional measured or manipulated variables; a good example is the dependence of d' on stimulus strength. Also, in a great majority of psychology studies in which classification tasks are used, the data have a hierarchical structure, i.e., there are repeated measures for each participant or item, and participants or items are only samples from the target population. A general-purpose

[1] In contrast to the formal assumptions, a psychological interpretation of the SDT parameters can be tested even when ratings are not available, e.g., by means of selective influence (Sternberg, 2001)

method of inference for SDT models should accommodate both kinds of situations.

The importance of hierarchical regression structure

If data have a hierarchical structure, but variability due to subjects, items, or other sampled factors is not accounted for, estimates of average (fixed) effects are not guaranteed to be unbiased and conclusions are not guaranteed to generalize to the target population. The not uncommon practice of analyzing data aggregated over sampled factors represents an extreme case of ignoring hierarchical data structure. The invalidity of this approach in the context of SDT was clearly illustrated with the results of simulational studies by Morey, Pratte, and Rouder (2008); however, strictly speaking, it requires such demonstrations only in order to illustrate a mathematical truth in a specific case. Because SDT is a non-linear model, by definition, when estimates of its parameters are based on data aggregated over sampled factors (e.g., d' estimated for hits and false alarms averaged over subjects), the expected values of these estimates (e.g., what the calculated d' actually estimates) are not in general true population averages (e.g., true d' in some condition). Such estimates are biased and inference about a target population based on them is simply not valid. To give a concrete example, consider two unbiased participants, one with $d'_1 = 2$ and one with $d'_2 = 4$. Their expected average accuracy is given by $(\Phi(d'_1/2) + \Phi(d'_2/2))/2 = .91$, which corresponds to $d' = 2.68$, whereas their true average d' is 3.

Aggregation is not the only way of ignoring a hierarchical data structure. Sometimes non-aggregated data are analyzed by using separate estimates for every participant × item × condition combination, but uncertainty due to distributions of subject or item effects is not accounted for by means of a hierarchical model structure. In such cases, conclusions – at least with respect to the uncertainties in estimates of population-average (fixed) effects are guaranteed to be valid only for the given sample, not the target population.

Another common source of problems is the separation of estimation of non-linear model parameters from regression analyses which aim to relate these parameters to measured or manipulated variables. When the SDT parameters are estimated separately for each subject and condition, and only later are these estimates regressed on predictors of interest, a number of additional issues may arise.

Firstly, the standard errors or credible intervals associated with the regression coefficients do not reflect the uncertainty in the SDT parameter estimates because the latter are treated as mere data points. The precision of parameter estimates often varies between participants, items, or conditions, but when the estimates are treated as data points, no use is made of this information. Secondly, regressing parameters on numeric predictors makes their estimates dependent on the common regression structure, and so also on each other, which can improve the quality of the estimates, just as assuming that random effects are samples from a common distribution may improve their estimates.

The aforementioned problems can be dealt with by supplementing the SDT model with the hierarchical linear regression structure. This can only be achieved if the model is substantially reparametrized.

Hierarchical Signal Detection Theory in a constrained parameter space

Both d' and c have the virtue of being directly interpretable in terms of sensitivity and bias. However, both d' and c are constrained: d' is non-negative and, when there is more than one criterion, the elements of the $\boldsymbol{c}$ vector are order restricted ($\boldsymbol{c}_{i+1} > \boldsymbol{c}_i$). Hierarchical linear regression structure can only be defined on unconstrained parameters, because random effects are assumed to be normally distributed and normal distribution is unbounded and because fixed effects are represented by unconstrained parameters. We provide examples of the problems that may arise when the SDT model is not correctly reparametrized in the following summary of two hierarchical SDT implementations.

We are aware of two failed attempts at creating a general-purpose method of inference for hierarchical SDT models with ratings. One is the Gibbs sampler proposed by Morey et al. (2008) and the other is the Hierarchical Meta-d' model (HMeta-d') proposed by Fleming (2017). The HMeta-d' model is a hierarchical version of the meta-d' model (Maniscalco & Lau, 2012), which in turn is a generalization of the SDT model that allows for a separate "meta-sensitivity" to account for possible discrepancies between a binary stimulus classification (referred to as a type 1 task) and the associated rating task (referred to as a type 2 or meta-cognitive task). We consider HMeta-d' here because it reduces to the SDT model with ratings when the type 1 and type 2 sensitivities are equal.

The Gibbs sampler created by Morey et al. (2008) allows for at most two sampled factors to have independent normally distributed random effects on the evidence distribution means. Unlike d', each evidence distribution mean considered in isolation is an unconstrained parameter, but the mean of the second evidence distribution is by definition greater than ($d' > 0$) or equal to ($d' = 0$) the mean of the first. The authors explicitly admit that their algorithm does not enforce this restriction because, as they claim in the paper, it would greatly complicate analysis. At the same time the authors fail to mention that an immediate consequence of allowing for negative d' values is that the resulting posterior distribution loses its intended interpretation since it can have a non-zero mass when $d' < 0$. The outermost criteria are fixed at 0 and 1, and the ordering restriction is enforced by assuming that the likelihood is 0 whenever $\boldsymbol{c}_{i+1} \leq \boldsymbol{c}_i$. As the authors explain, because a sampled factor can have independent random effects on the evidence distribution means, it can have an effect on all the criteria: shifting both means by the same amount in the same direction is equivalent to keeping the sensitivity intact, while shifting the criteria relative to the evidence distributions. However, the elements of the criteria vector cannot be affected differently by the same sampled factor, which is an unrealistic restriction. Participants differ in how they place the criteria just as they differ in their sensitivity and it is quite impossible for all the participants to place the criteria in the same distance from each other.

In HMeta-d' the hierarchical structure is restricted to normally distributed random intercepts of one sampled factor. In the HMeta-d' model the d' parameter is allowed to assume negative values also, but the most problematic aspect of this implementation is again the representation of the criteria. Each element of the criteria vector has an associated independent normal distribution, which does allow for criteria random effects, but does not enforce the necessary ordering restriction. The elements of this vector are sorted to obtain another criteria vector, and it is this sorted criteria vector that is used to model the conditional combined response distributions. Consequently, the model does contain parameters representing the actual, order-restricted criteria, but, because sorting is not injective, the space of the actual criteria is only loosely related (i.e., not isomorphic) to the space of the unrestricted criteria vectors that are associated with the hierarchical structure.

Hierarchical Signal Detection Theory in an unconstrained parameter space

The general hierarchical linear regression structure can be defined on SDT parameters only if the latter are derived from unconstrained parameters. In the `bhsdtr` package, d' is derived from $\delta = \ln(d')$, thus random effects on d' can be modelled by assuming that δ is normally distributed. The problem of representing the criteria by unconstrained parameters is solved by mapping the R^{K-1} space of unconstrained criteria vectors to the K dimensional probability simplex space using the softmax function, and mapping the simplex space to the space of order-restricted criteria vectors by means of the inverse normal CDF:

$$\boldsymbol{c}_i = \Phi^{-1}(\sum_{k=1}^{i}(e^{\gamma_k})/\sum_{j=1}^{K}(e^{\gamma_j})) \tag{1}$$

where Φ is the CDF of the standard normal distribution and $\boldsymbol{\gamma} \in R^K$, with $\boldsymbol{\gamma}_K$ fixed at 0 for identifiability. The idea is illustrated in Fig. 3 below:

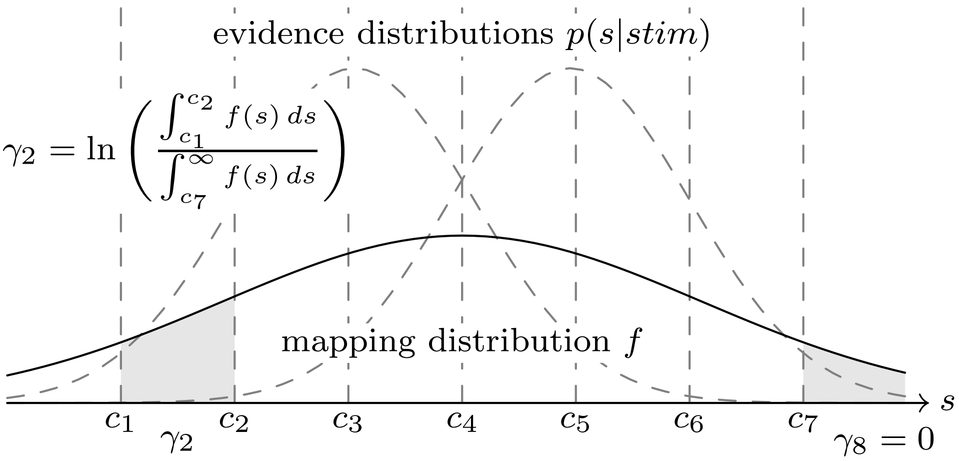
evidence distributions $p(s|stim)$
$\gamma_2 = \ln\left(\frac{\int_{c_1}^{c_2} f(s)\,ds}{\int_{c_7}^{\infty} f(s)\,ds}\right)$
mapping distribution f
s
c_1
c_2
c_3
c_4
c_5
c_6
c_7
γ_2
$\gamma_8 = 0$

Figure 3. Mapping between the unconstrained $\boldsymbol{\gamma}$ vector and the criteria

Note that the normal distribution centered at the midpoint is merely a mapping device, not a third evidence distribution, and that, for the reasons that will soon become clear, it is wider than the two evidence distributions. The mapping expressed by Eq. 1 is an isomorphism between the R^{K-1} space and the space of order-restricted criteria vectors. It's inverse is given by $\gamma_i = \ln\left(\int_{c_{i-1}}^{c_i} f(s)\,ds / \int_{c_{K-1}}^{\infty} f(s)\,ds\right)$, where f is the standard normal probability density function. The elements of the $\boldsymbol{\gamma}$ vector correspond to relative distances between pairs of adjacent criteria because their exponents represent the relative magnitudes of areas under the standard normal curve, delineated by the pairs of adjacent criteria: $e^{\gamma_i}/e^{\gamma_j} = (\Phi(\boldsymbol{c}_i) - \Phi(\boldsymbol{c}_{i-1}))/(\Phi(\boldsymbol{c}_j) - \Phi(\boldsymbol{c}_{j-1}))$. When $K = 2$, only $\boldsymbol{\gamma}_1$ is free to vary, and its value directly represents the direction and magnitude of bias: $\boldsymbol{\gamma}_1$ is 0 when the criterion is placed at the midpoint between the evidence distributions; the more negative (positive) $\boldsymbol{\gamma}_1$ is, the more the criterion is shifted to the left (right) of the midpoint.

It is often a good idea to multiply all the criteria by a value greater than 1, which is equivalent to making the mapping distribution wider. This tends to even out values of $\boldsymbol{\gamma}$ by preventing the outermost areas under the mapping distribution curve from becoming very small relative to ares delineated by adjacent pairs of non-outermost criteria. This is especially important when the criteria are widely spread, as can happen for moderate to large d' values. This feature is implemented in the `bhsdtr` package by introducing a criteria scaling factor.

Once d' and c are derived from the unconstrained δ and γ parameters, the SDT model can be supplemented with a hierarchical linear regression structure. To avoid having to deal with an even more complicated index notation[2], below we present only the simple case of one sampled factor.

$$
\begin{aligned}
\boldsymbol{\delta} &= \boldsymbol{X}^{(\delta)}\boldsymbol{\beta}^{(\delta)} + \boldsymbol{Z}^{(\delta)}\boldsymbol{\theta}^{(\delta)} \\
\boldsymbol{d'_i} &= e^{\boldsymbol{\delta}_i} \\
\boldsymbol{\gamma}_{i,\cdot} &= \boldsymbol{X}^{(\gamma)}_{i,\cdot}\boldsymbol{\beta}^{(\gamma)} + \boldsymbol{Z}^{(\gamma)}_{i,\cdot}\boldsymbol{\theta}^{(\gamma)} \\
\boldsymbol{c}_{i,k} &= s\ \Phi^{-1}(\sum_{l=1}^{k}(e^{\gamma_{i,l}})/\sum_{m=1}^{K}(e^{\gamma_{i,m}})) \\
p(\boldsymbol{y}_i = k|\boldsymbol{stim}_i = 1) &= \Phi(\boldsymbol{c}_{i,k} + \boldsymbol{d'_i}/2) - \Phi(\boldsymbol{c}_{i,k-1} + \boldsymbol{d'_i}/2) \\
p(\boldsymbol{y}_i = k|\boldsymbol{stim}_i = 2) &= \Phi(\boldsymbol{c}_{i,k} - \boldsymbol{d'_i}/2) - \Phi(\boldsymbol{c}_{i,k-1} - \boldsymbol{d'_i}/2)
\end{aligned}
$$

Here $i = 1 \ldots N$ is the observation number, $\boldsymbol{X}$ is the fixed effects model matrix for the respective parameter, $\boldsymbol{Z}$ is the random effects model matrix, $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$ are the fixed and random effects, $\boldsymbol{c}$ is an $N \times K - 1$ matrix, s is the criteria scaling factor, and $\boldsymbol{y}$ is the combined response. Note that $\boldsymbol{d'_i}$ is a scalar, but $\boldsymbol{\gamma}_{i,\cdot}$ is in general a vector, and so $\boldsymbol{\beta}^{(\gamma)}$ and $\boldsymbol{\theta}^{(\gamma)}$ are matrices. The j-th rows of the $\boldsymbol{\beta}^{(\gamma)}$ and $\boldsymbol{\theta}^{(\gamma)}$ matrices represent fixed and random effects on the j-th element of the $\boldsymbol{\gamma}$ vector.

Following Sorensen and Vasishth (2015) we make use of the Cholesky decomposition of the correlation matrices because it improves efficiency and admits a convenient prior on random effects correlations:

[2] The reader familiar with hierarchical models may be surprised by our use of superscript parenthesized Greek letters to express hierarchical relationships. We chose this convention because it allowed us to use subscripts to denote elements of vectors and matrices while minimizing the number of nested sub- or superscripts.

$$\begin{aligned}
\text{vectorized}(\boldsymbol{\theta}^{(\gamma)}) &= \text{diag}(\boldsymbol{\tau}^{(\gamma)})\boldsymbol{L}^{(\gamma)}\boldsymbol{z}^{(\gamma)} \\
\boldsymbol{\theta}^{(\delta)} &= \text{diag}(\boldsymbol{\tau}^{(\delta)})\boldsymbol{L}^{(\delta)}\boldsymbol{z}^{(\delta)} \\
\boldsymbol{z}_i^{(\delta)} &\sim \text{Normal}(0, 1) \\
\boldsymbol{z}_j^{(\gamma)} &\sim \text{Normal}(0, 1)
\end{aligned}$$

where each $\boldsymbol{\tau}$ is a vector of standard deviations of random effects and each $\boldsymbol{L}$ is a Cholesky decomposition of a random effects correlation matrix, i.e., $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}'$. Thus, $\boldsymbol{\theta}$ is multivariate normal with covariance matrix diag$(\boldsymbol{\tau})\boldsymbol{L}$.

Finally, we use weakly informative proper priors because they provide regularization and help stabilize computation. The fixed effects $\boldsymbol{\beta}^{(\delta)}$ and $\boldsymbol{\beta}^{(\gamma)}$ are given independent normal priors, the random effects standard deviations $\boldsymbol{\tau}^{(\delta)}$ and $\boldsymbol{\tau}^{(\gamma)}$ are given independent half-Cauchy priors, as recommended by Gelman (2004), and each $\boldsymbol{L}$ is given an independent lkj prior:

$$\begin{aligned}
\boldsymbol{\beta}_i^{(\delta)} &\sim \text{Normal}(\boldsymbol{\mu}_i^{(\delta)}, \boldsymbol{\sigma}_i^{(\delta)}) \\
\boldsymbol{\beta}_{k,l}^{(\gamma)} &\sim \text{Normal}(\boldsymbol{\mu}_{k,l}^{(\gamma)}, \boldsymbol{\sigma}_{k,l}^{(\gamma)}) \\
\boldsymbol{\tau}_i^{(\delta)} &\sim \text{half-Cauchy}(0, \boldsymbol{\zeta}_i^{(\delta)}) \\
\boldsymbol{\tau}_{k,l}^{(\gamma)} &\sim \text{half-Cauchy}(0, \boldsymbol{\zeta}_{k,l}^{(\gamma)}) \\
\boldsymbol{L}^{(\delta)} &\sim \text{lkj}(\nu^{(\delta)}) \\
\boldsymbol{L}^{(\gamma)} &\sim \text{lkj}(\nu^{(\gamma)})
\end{aligned}$$

Specifying the prior distributions

The formal definition of a Bayesian model is not complete without providing fixed values of all the parameters that define prior distributions. Specifying the priors on sensitivity effects does not pose any special difficulties. The sensitivity of an unbiased classifier given percent correct (pc) is given by $2\Phi^{-1}(\text{pc})$. When $p(stim = 1) = 0.5$, the greater the bias, the lower the accuracy, meaning that an unbiased sensitivity is a lower bound on sensitivity given percent correct. Let us assume that the majority of subjects are expected to achieve percent correct within the .51 to .99 range, with negligible bias. Since $\ln(2\Phi^{-1}(.51)) = -2.99$ and $\ln(2\Phi^{-1}(.99)) = 1.54$, a reasonable weakly informative prior on δ is normal with mean $(1.54 - 2.99)/2$ and standard deviation $(1.54 + 2.99)/2$, which is the default prior on delta effects in the `bhsdtr` package.

Specifying the priors on criteria effects can be challenging because the criteria are order-restricted and the complexity of the mapping expressed by Eq. 1 makes it difficult to reason about priors on $\boldsymbol{\gamma}$ in terms of criteria effects. By default, in the `bhsdtr` package each entry in the $\boldsymbol{\sigma}^{(\gamma)}$ and $\boldsymbol{\zeta}^{(\gamma)}$ matrices is set to $\ln(100)$ and the criteria scaling factor is fixed at 2.

The prior on random effects standard deviations is parametrized by $\boldsymbol{\zeta}$, which represents half-width at half-maximum of the half-Cauchy distribution. In our opinion, a not-unreasonable starting point is to set $\boldsymbol{\zeta}$ at the value that is greater or equal to the most likely value of the random effects standard deviation.

Finally, by default $\nu^{(\delta)} = \nu^{(\gamma)} = 1$, which implies a uniform prior on random effects correlation matrices. Because the greater the value of ν, the more emphasis is put on zero off-diagonal correlations, the researcher can force the correlations to be near-zero by choosing a large ν value.

Overview of the software implementation

The `bhsdtr` package implements the model described in the previous section in the Stan modelling language because it uses a state-of-the-art adaptive Hamiltonian Monte Carlo sampling algorithm which often handles high-dimensional correlated posteriors better than a Gibbs sampler. Our package is essentially a collection of documented functions: The `aggregate_responses` function aggregates data as much as possible for efficiency, but without distorting the hierarchical structure. The `make_stan_model` function creates a model definition in the Stan language. The Stan code produced by the `make_stan_model` function can be fitted as is or modified by the user if needed, e.g., to change the prior distributions or to drop the equal variance assumption. The `make_stan_data` function creates regression model matrices and other data structures required by the model created using the `make_stan_model` function. Finally, the `plot_sdt_fit` function can be used to visually assess the fit of the model by creating publication-ready ROC curve plots or response distribution plots with posterior predictive intervals calculated for the chosen α level.

Usage example: installing the package and testing the model on real data

To make full use of the bhsdtr package functionality, three non-standard R packages are required, namely `rstan`, `plyr`, and `ggplot2`. We recommend using the `devtools` package to install the `bhsdtr` package directly from the github repository. This will automatically install any missing required packages:

```
devtools::install_git('git://github.com/boryspaulewicz/bhsdtr')
library(bhsdtr)
```

We will now explain how to perform some of the essential steps of a typical data analysis process, which at the very least will usually involve preparing the data, creating the model code, fitting the model, assessing the fit, and possibly converting the unconstrained δ and γ parameters to d' and c.

Preparing the data. The `bhsdtr` package contains a dataset, `gabor`, from an unpublished study in which on each trial the participants had to classify a briefly presented Gabor patch as tilted to the left or to the right using the arrow keys. The participants were also asked to rate the stimuli on a 4-point Perceptual Awareness Scale (Ramsøy & Overgaard, 2004) presented at the bottom of the screen. The Gabor patch was immediately followed by a mask. The PAS ratings ranged from "no experience" to "absolutely clear image" and were provided either before (RD order condition) or after (DR order condition) the arrow keys were pressed. On each trial the Gabor patch was equally likely to be presented for 32 ms or 64 ms. Order was a between-subject variable and duration was a within-subject variable. There were 47 participants and 48 trials per condition.

In the study in question, the response was originally encoded using separate variables for accuracy and rating, so the first step was to create an appropriate response variable using the `combined_response` function. This function requires three variables, one encoding the stimulus class, one encoding the rating (as an integer), and one binary variable encoding the decision accuracy.

```
  gabor$resp = combined_response(gabor$stim,
                                 gabor$rating,
                                 gabor$acc)
```

This step is required only if the ratings are available and a combined response variable is not already present in the data. In the single criterion case, the combined response variable is simply the binary classification decision. To fit a single-criterion SDT model to this dataset, the code above would have to be replaced with the following:

```
  gabor$resp = combined_response(gabor$stim,
                                 accuracy = gabor$acc)
```

Next, the data has to be aggregated using the `aggregate_responses` function, but only to an extent that preserves all the random effects. This function requires as arguments a data frame containing all the relevant variables, the name of the stimulus class variable, the name of the combined response variable, and the vector of the names of all the variables that are to be preserved in the resulting aggregated dataset (apart from the stimulus class variable and the combined response variable), i.e., those encoding the sampled factors and those representing the independent variables used in the regression part of the model:

```
adata = aggregate_responses(gabor, 'stim', 'resp',
                            c('id', 'duration', 'order'))
```

The main purpose of the aggregation step is to improve the efficiency of sampling from the posterior distribution. When data are aggregated in this way, the likelihood for each condition × participant combination has to be computed only once rather than as many times as there are trials per condition per participant. Note that if there are other sampled factors present in the data (e.g., items, replications, etc.), then these factors also have to be specified at this stage to preserve the hierarchical data structure. The `aggregate_responses` function creates a list with three components. The `data` component is a data frame containing additional preserved variables, the `stimulus` component is the stimulus class variable, and the `counts` component is an $N \times K$ matrix of combined response counts, where N is the number of data points and K is the number of possible combined response values.

Creating the model code. A model is fitted using the `stan` function from the `rstan` package. This function requires a special list of data structures used by the model as well as a model specification expressed in the Stan language. Every model has some fixed effects structure since, even when there are no predictors, the model parameters can be expressed as regressed on a vector of ones (i.e., an intercept). However, many models also have a hierarchical structure and, if that is the case, this hierarchical structure has to be specified when using the `make_stan_model` function. This is done by providing a list of lists of R model formulae. Each list of model formulae is composed of at least three elements and specifies the correlated random effects of one sampled factor. The `group` element specifies the sampled factor; the `delta` and `gamma` elements specify which effects are assumed to vary between the levels of this sampled factor. When `make_stan_model` is used without any arguments, it specifies a model without any random effects. Fixed effects model matrices are specified by providing a list with at least two model formulae, named `delta` and `gamma`, to the `make_stan_data` function that is described later in this paper. Non-default priors can be specified by adding optional elements to the random and fixed effects specification lists, as described in the `make_stan_data` function documentation.

In the study in question there was only one sampled factor, i.e., the participants. Because duration was a within-participant variable, in principle its effect could vary between the participants for all the SDT parameters. However, a preliminary data analysis indicated that the barely noticeable 32 ms difference in duration seemed to affect only the sensitivity. Thus, it was assumed that δ may depend on duration and order, but γ may only be affected by order. Because duration was a within-subjects variable, its effect on δ was assumed to vary between the participants, but the only random effect associated with γ was the intercept:

```
fixed = list(delta = ~ -1 + duration:order,
             gamma = ~ order)
random = list(list(group = ~ id,
                   delta = ~ -1 + duration,
                   gamma = ~ 1))
model = make_stan_model(random)
```

The `make_stan_data` function creates fixed and random effect model matrices based on the respective model formulae using dummy contrast coding. Note that the implicit intercept was suppressed for the δ model matrix. In this way, δ was estimated for every duration × order condition. The resulting separate intercepts and slopes parametrization makes it easier to calculate arbitrary contrasts on posterior samples. A more standard parametrization was used for the γ parameter because it was initially assumed that the criteria depend only on order, and so there was only one contrast of interest for every element of the γ vector. On the other hand, in such cases nested parametrization (with separate intercepts and slopes for every condition) may be more convenient if a researcher is interested in the actual criteria, as we will later explain when introducing the `gamma_to_crit` function. This example also illustrates how the separation of the δ and γ regression structures makes it possible to test a broad class of linear models representing the dependence of the SDT parameters on additional variables.

Fitting the model. In order to fit the model, a separate data structure used by the Stan sampler has to be created using the `make_stan_data` function. The obligatory arguments to this function are an aggregated data object created by the `aggregate_responses` function and a fixed effects specification. Importantly, if random effects are modelled, the same specification of random effects has to be provided to the `make_stan_model` and `make_stan_data` functions.

```
sdata = make_stan_data(adata, fixed, random)
```

Finally, a vector of names of parameters of interest has to be specified when calling the `stan` function:

```
fit = stan(model_code = model,
           data = sdata,
           pars = c('delta_fixed', 'gamma_fixed',
                    'delta_sd_1', 'gamma_sd_1',
                    'delta_random_1', 'gamma_random_1',
                    'Corr_delta_1', 'Corr_gamma_1',
                    'counts_new'),
           iter = 8000,
           chains = 4)
```

Note that since more than one sampled factor is allowed, the names of all the hierarchical parameters are indexed (e.g., `delta_sd_1`, `delta_random_1`, `Corr_delta_1`). The name `counts_new` refers to posterior predictive samples that are required by the `plot_sdt_fit` function. Names starting with `Corr` refer to random effects correlation matrices, which are computed from Cholesky decompositions.

Assessing the model fit. Four chains of 8,000 iterations each were run simultaneously; the first half of the posterior samples, which served as a warm-up period for tuning the parameters of the sampling algorithm, was discarded. Part of the resulting Stan output is presented in Table 1 below.

Table 1

Model fit summary statistics

	mean	SE_{mean}	SD	2.5%	97.5%	No. eff. samples	$\hat{R}$
delta_fixed[1]	-0.10	0.00	0.15	-0.41	0.18	4626	1.00
delta_fixed[2]	1.12	0.00	0.09	0.95	1.29	4764	1.00
delta_fixed[3]	-0.39	0.00	0.20	-0.80	-0.03	5427	1.00
delta_fixed[4]	1.28	0.00	0.11	1.06	1.50	5464	1.00
gamma_fixed[1,1]	-0.14	0.00	0.06	-0.27	-0.02	2925	1.00
gamma_fixed[1,2]	-0.22	0.00	0.10	-0.42	-0.01	8199	1.00
gamma_fixed[2,1]	-0.70	0.00	0.18	-1.05	-0.35	3060	1.00
gamma_fixed[2,2]	0.49	0.00	0.29	-0.06	1.04	3452	1.00
gamma_fixed[3,1]	-0.54	0.00	0.22	-0.96	-0.11	3103	1.00
gamma_fixed[3,2]	0.83	0.01	0.35	0.14	1.51	3086	1.00
gamma_fixed[4,1]	0.28	0.00	0.25	-0.20	0.76	3090	1.00
gamma_fixed[4,2]	0.42	0.01	0.40	-0.37	1.22	3291	1.00
gamma_fixed[5,1]	-0.21	0.01	0.30	-0.80	0.37	3440	1.00
gamma_fixed[5,2]	0.83	0.01	0.48	-0.10	1.78	3637	1.00
gamma_fixed[6,1]	-0.77	0.00	0.24	-1.23	-0.31	3378	1.00
gamma_fixed[6,2]	0.80	0.01	0.38	0.05	1.56	3316	1.00
gamma_fixed[7,1]	-0.32	0.00	0.16	-0.64	0.01	3346	1.00
gamma_fixed[7,2]	0.42	0.00	0.28	-0.11	0.98	3333	1.00
delta_sd_1[1]	0.66	0.00	0.11	0.48	0.90	6373	1.00
delta_sd_1[2]	0.44	0.00	0.05	0.35	0.56	5463	1.00
gamma_sd_1[1]	0.16	0.00	0.08	0.02	0.31	2071	1.00
gamma_sd_1[2]	0.82	0.00	0.11	0.62	1.06	5852	1.00
gamma_sd_1[3]	1.08	0.00	0.12	0.86	1.33	4293	1.00
gamma_sd_1[4]	1.27	0.00	0.14	1.03	1.57	3723	1.00
gamma_sd_1[5]	1.55	0.00	0.17	1.24	1.91	4297	1.00
...	...	...	...	...	...	...	...

As can be seen, given the complexity of the model, the chains exhibited good mixing and seemed to have converged; there were enough effective samples of the fixed effect parameters to estimate 95% credible intervals well and none of the Gelman-Rubin $\hat{R}$ statistics crossed the conventional 1.01 threshold, suggesting negligible sensitivity to the initial values.

Figs. 4 and 5 below contain normal quantile-quantile plots of random effects. The plots indicate that the normally distributions of random δ and γ effects can be approximated by normal distributions and that these parameters seem to be good candidates for representing variability in the sensitivity and criteria parameters due to the sampled factors.

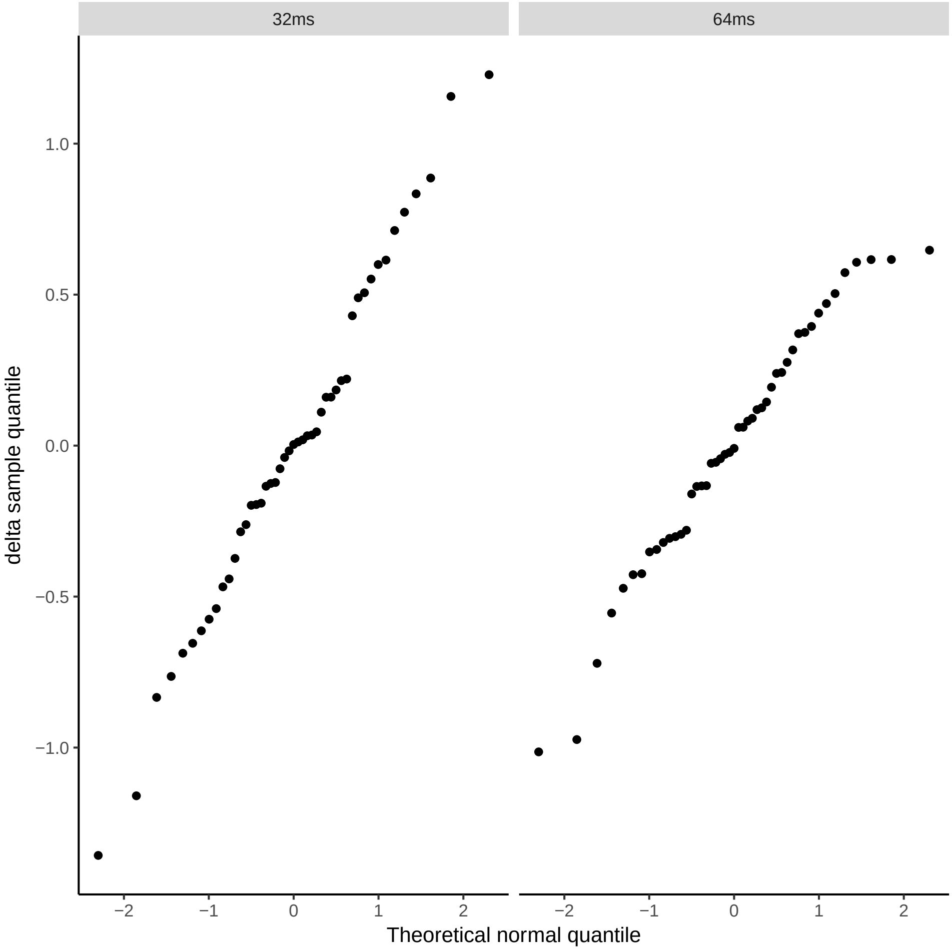

Figure 4. Normal quantile-quantile plots of δ random effects

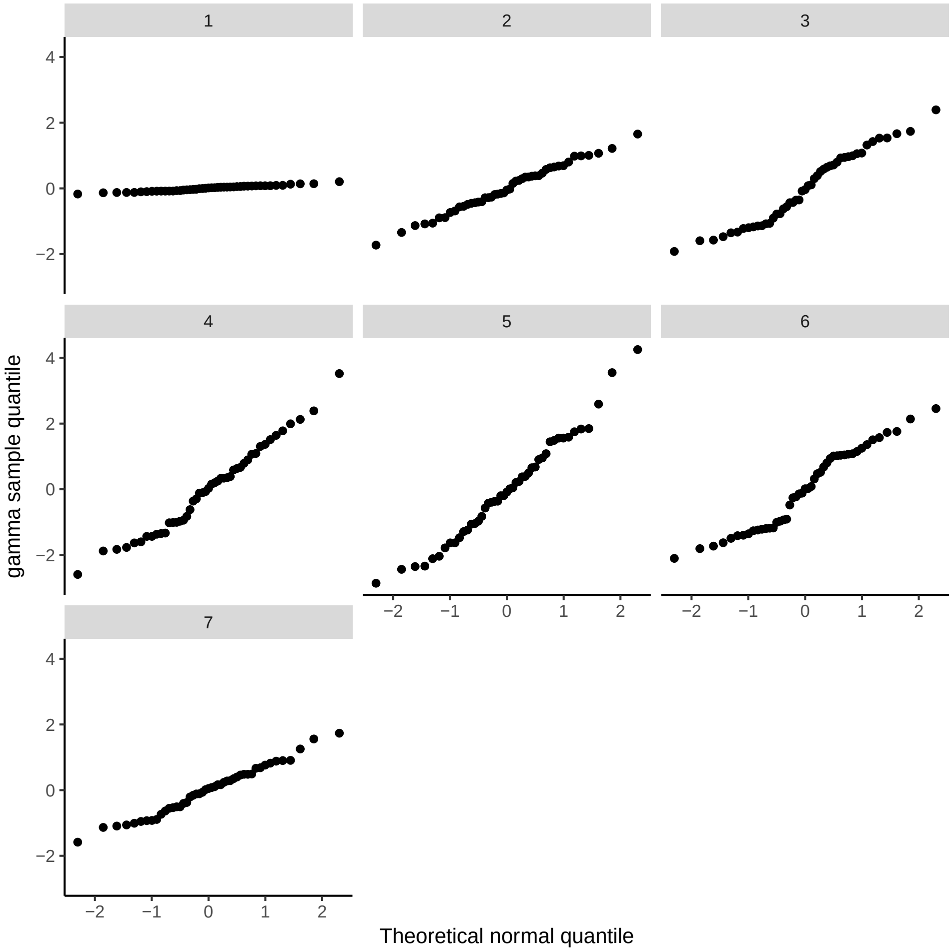

Figure 5. Normal quantile-quantile plots of γ random effects

Once enough good quality posterior samples are obtained for the parameters of interest, the inference process can be carried out by calculating credible intervals, HPD intervals, or Bayes factors for any function of the parameters. However, even when the stan output summary does not indicate sampler convergence issues, before drawing any further conclusions the researcher should first check if the model fits the data. The `plot_sdt_fit` function can be used for this purpose:

```
plot_sdt_fit(fit, adata, c('order', 'duration'))
```

This function requires at least three arguments: a `stanfit` object, an aggregated data list produced by the `aggregate_responses` function that was used to produce the stanfit object, and a vector of names of variables that will determine how the data will be partitioned before plotting. We recommend assessing the fit at the individual level, but we did not include the participant identification number in the list of conditioning variables because the resulting plot would take up too much space.

As can be seen in Fig. 6, which shows the ROC curves produced by the code above, the model seemed to fit the data well in all but one condition (decision after rating, 64 ms duration), in which three out of seven relevant[3] points were outside the two-dimensional 95% posterior predictive regions.

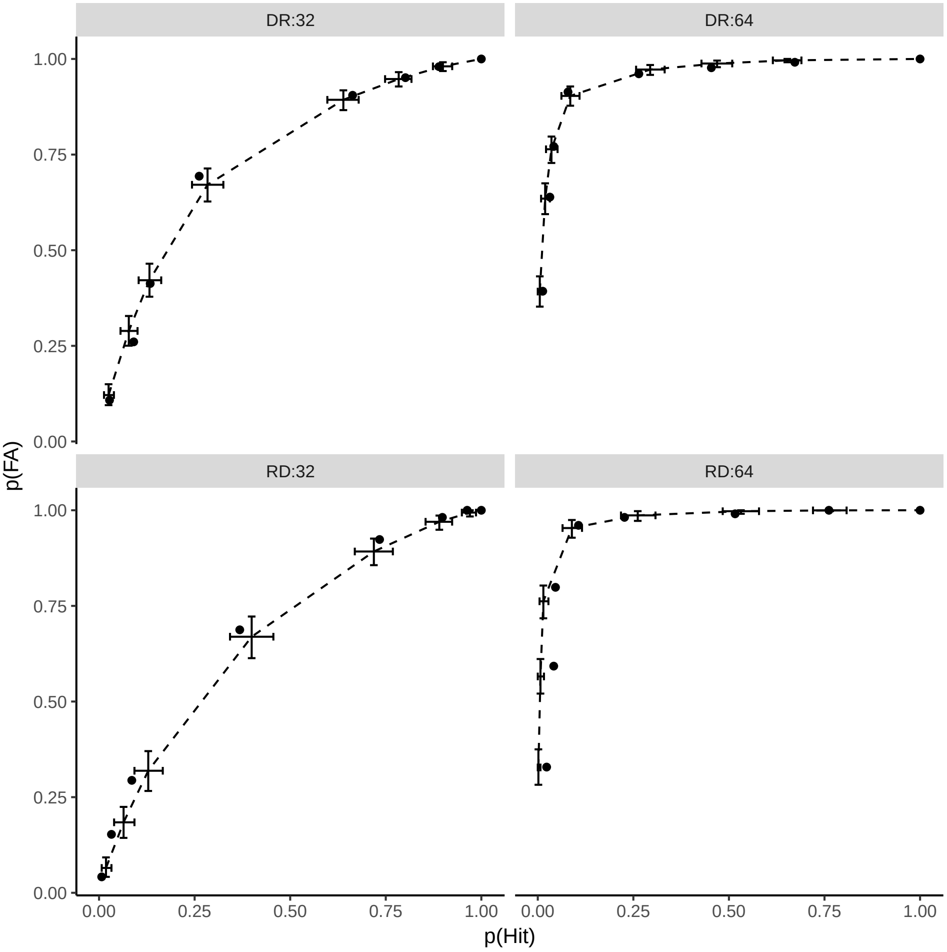

Figure 6. ROC curve fit

Another way to assess model fit visually is by inspecting the conditional response distributions ($p(y|stim)$), such as those shown in Fig. 7, which was also created using the `plot_sdt_fit` function.

[3] The point in the upper-right corner of a ROC curve is always in the $(1, 1)$ position

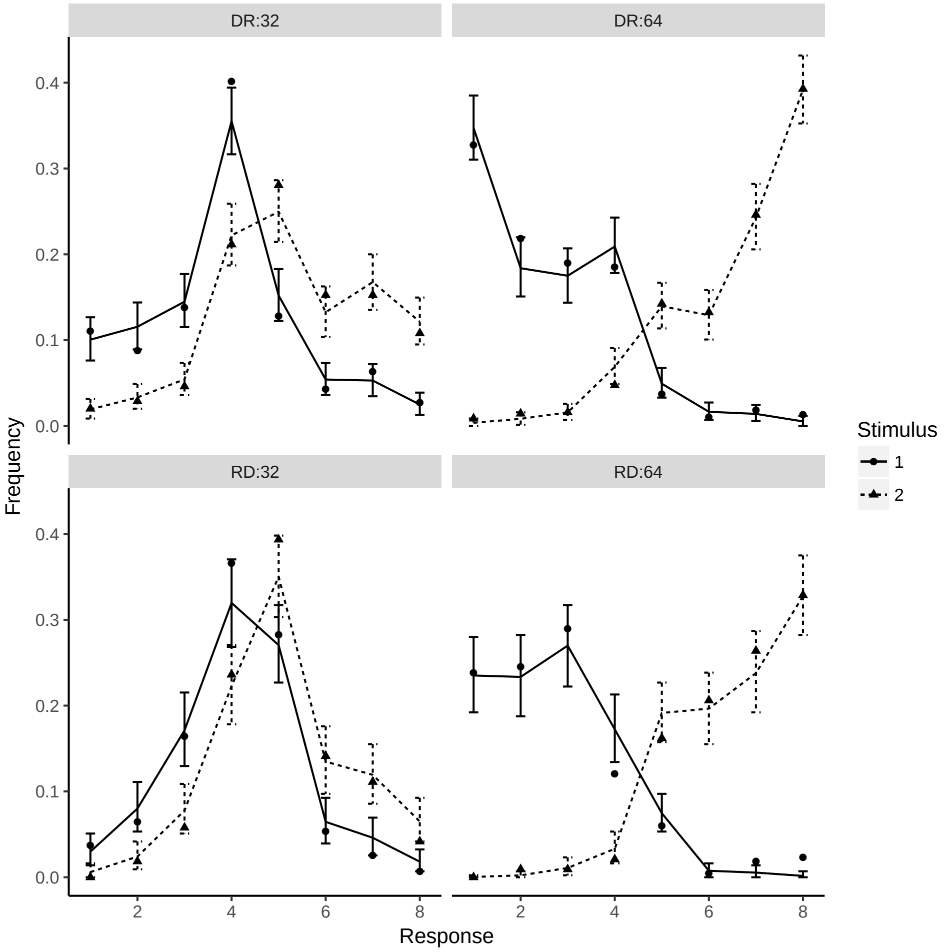

Figure 7. Response distribution fit

This kind of plot can be informative about the reasons why a model does not fit the data. In this particular case, the plot seems to suggest that it may be a good idea to inspect the fit at the individual level and see if there are some participants with unusual $p(y|stim = 1)$ distributions in the RD × 64 ms condition. On the other hand, it is also possible that the lack of fit is mainly a consequence of the assumption that duration had zero effect on $\boldsymbol{\gamma}$, or that more substantial modifications are necessary, such as dropping the equal variance assumption.

Converting unconstrained δ and γ parameters to sensitivities and criteria. Posterior δ and γ samples have to be transformed in order to work with the d' and c parameters. This is straightforward only when fixed effects represent average parameter values in separate conditions, not differences between conditions or regression slopes. In our example, because nested parametrization was used for the δ fixed effects model matrix, all four `delta_fixed` parameters can easily be transformed to sensitivities by applying the exponential function. It is important to remember that because this is a non-linear transformation, the δ to d' conversion step should be done first before applying any other transformations to the posterior samples; for example, the exponential of a point and interval estimate is not equal to the point and interval estimate calculated after transforming the δ posterior samples to the d' samples.

In this case the first column of the `gamma_fixed` parameter matrix (the intercept) corresponds to the values of the $\boldsymbol{\gamma}$ vector in the DR condition, but the second column corresponds to the *effect* of order on $\boldsymbol{\gamma}$. For this reason, in this particular case the posterior criteria samples can be obtained using the `gamma_to_crit` function only for the first column of the `gamma_fixed` matrix.

Testing the model on simulated data

To test if the model correctly recovers known parameter values, we simulated the data from a hypothetical exact replication of the previously described experiment using the point estimates from the previous fit as known realistic parameter values. Mixing performance was similar to the real data case. All the model parameters were correctly recovered in a sense that the true values were outside the 95% credible intervals no more than 5% of the time.

For illustration purposes, an SDT model that differed from the true model only in that it did not have any hierarchical structure was fitted to the same simulated dataset. Since the non-hierarchical model was much simpler and the data consisted of only eight vectors of response counts, the mixing of the chains was excellent.

The 95% credible intervals calculated for the fixed effects based on each model are compared in Fig. 8 below. The estimates were centered on the true values to simplify the presentation. As can be seen, the true model correctly recovered the known parameter values, but the estimates based on the simplified, non-hierarchical model were biased; the credible intervals were not only about three times shorter on average, but also failed to contain most of the true values.

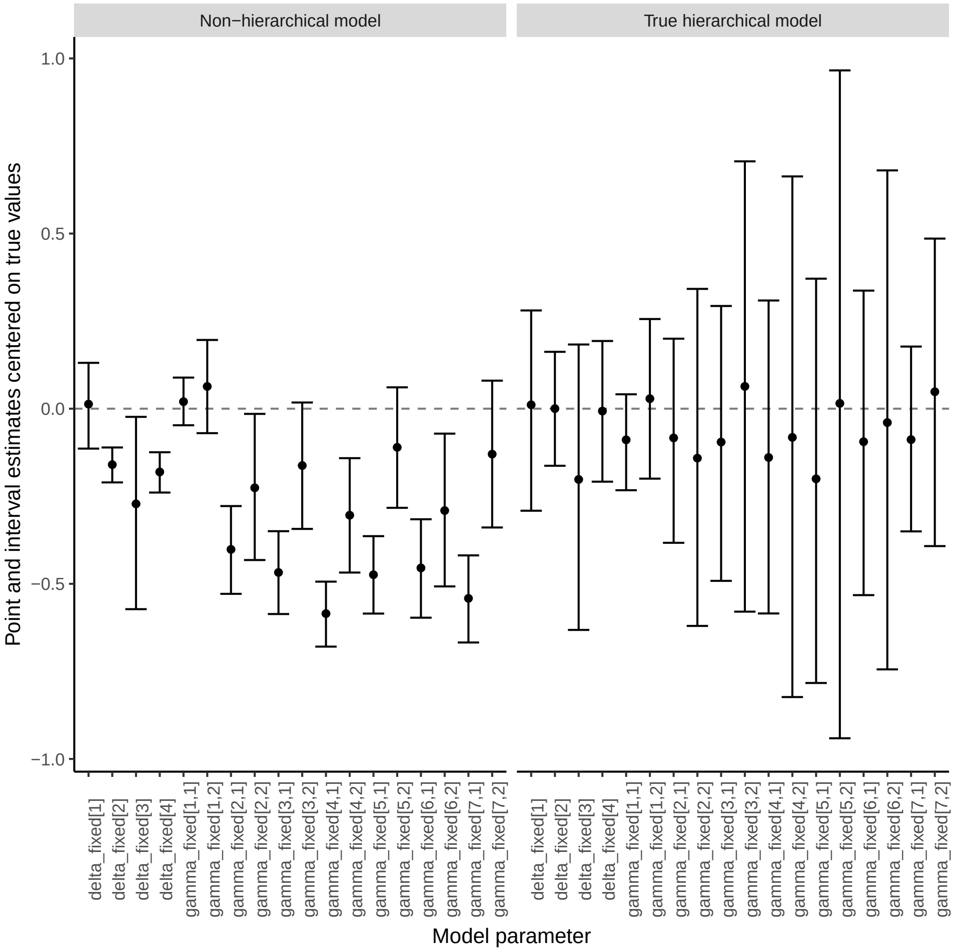

Figure 8. Comparison of the point and interval posterior estimates based on the true hierarchical and the simplified non-hierarchical models

However, as can be seen in Fig. 9 below, in this case the observed ROC curves seemed to fit the simplified model's predictions quite well, giving a false impression of model validity.

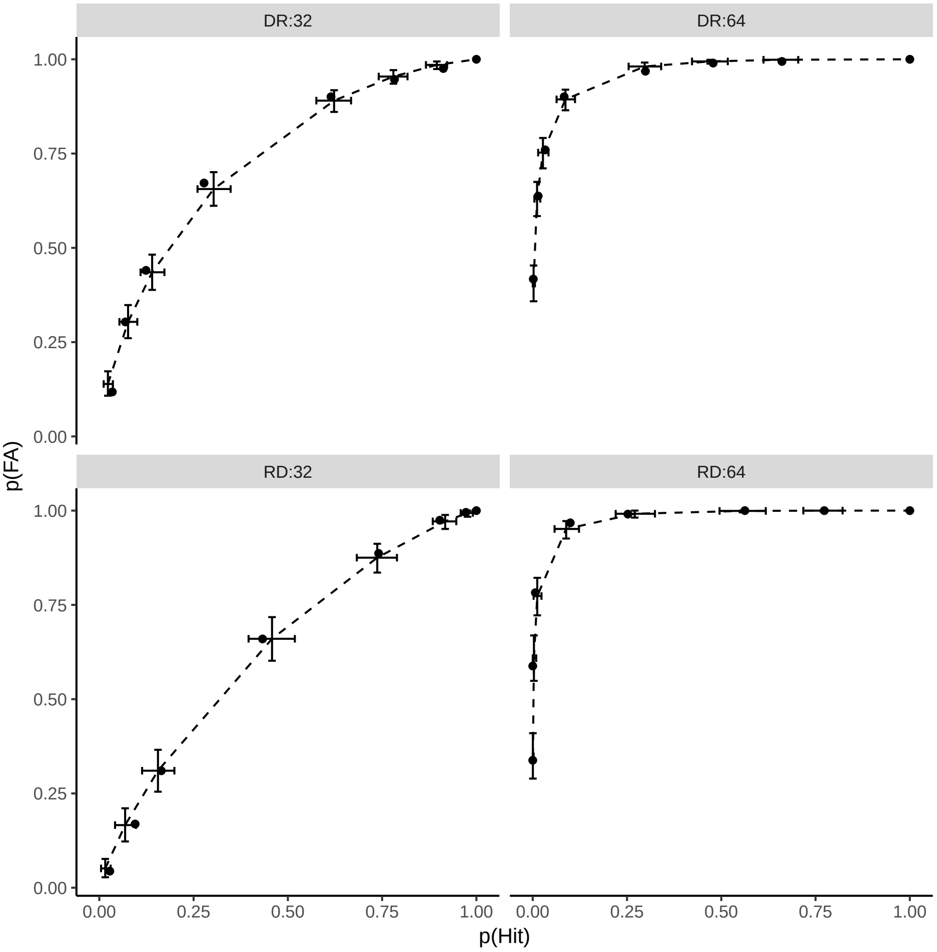

Figure 9. ROC curve fit for the non-hierarchical model

Concluding remarks

The great importance of SDT to psychology stems from the fact that given weak assumptions about an underlying decision process, it promises to deconfound sensitivity from bias in arbitrary binary classification tasks. To the best of our knowledge, at present the `bhsdtr` package provides the only method of Bayesian inference for SDT models with or without ratings that can be recommended as a default choice in typical applications. Our parametrization forces the sensitivity to be non-negative and the criteria to be order-restricted, while the isomorphisms between the d' and c parameters and the unconstrained δ and γ parameters make it possible to supplement the SDT model with the general hierarchical linear regression structure. There is no limit to the number of sampled factors except for the one imposed by available computational resources; correlations of random effects of the same sampled factor are accounted for, all the SDT parameters can be modelled by linear regression within the same model, and all the effects on all the SDT parameters estimable within the levels of the sampled factors can have associated random effects. If the need arises to relax a built-in restriction, experienced users can extend the model in arbitrary ways by using automatically generated human-readable Stan code as a template. We hope that researchers with a basic understanding of Signal Detection Theory, Bayesian inference, and hierarchical modelling will find our package useful and adopt it as a method of analysis of binary classification performance in future studies.

References

Fleming, S. M. (2017). HMeta-d: hierarchical bayesian estimation of metacognitive efficiency from confidence ratings. *Neuroscience of Consciousness*, *3*(1), nix007.

Gelman, A. (2004). Prior distributions for variance parameters in hierarchical models. *Bayesian Analysis*.

Green, D. M., & Swets, J. A. (1966). *Signal Detection Theory and Psychophysics.* New York: Wiley.

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

Morey, R. D., Pratte, M. S., & Rouder, J. N. (2008). Problematic effects of aggregation in z ROC analysis and a hierarchical modeling solution. *Journal of Mathematical Psychology*, *52*(6), 376–388.

R Core Team. (2017). R: A language and environment for statistical computing [Computer software manual]. Vienna, Austria. Retrieved from `https://www.R-project.org/`

Ramsøy, T. Z., & Overgaard, M. (2004). Introspection and subliminal perception. *Phenomenology and the Cognitive Sciences*, *3*(1), 1–23.

Sorensen, T., & Vasishth, S. (2015). Bayesian linear mixed models using stan: A tutorial for psychologists, linguists, and cognitive scientists. *arXiv preprint arXiv:1506.06201*.

Sternberg, S. (2001). Separate modifiability, mental modules, and the use of pure and composite measures to reveal them. *Acta psychologica*, *106*(1), 147–246.