

Table of Contents	
Data access workflow	2
Power analysis	4
Response Distributions of Cognitive Tasks	5
Overview of DDM Modeling Procedure	6
Imputation of the Mental Rotation Task	8
DDM simulations: The effect of few trials per participant	8
DDM simulation 1: Single condition with eight trials	9
DDM simulation 2: Two conditions; Boundary Separation fixed across conditions —	11
DDM simulation 3: Two conditions; 10,000 subjects	13
Additional DDM simulations	14
DDM simulation 4: Does shrinkage bias the associations between parameter estimates and adversity?	14
DDM simulation 5: Imputation of missing RTs	17
DDM Model Fit Assessments	18
Parameter recovery	18
Model convergence	21
Parameter distributions	23
SEM Fit	25

References	27
------------------	----

Data access workflow

Prior to Stage 1 submission of the Registered Report, we accessed the cognitive task data for a couple of [preregistered](#) data checks. By only accessing the cognitive task data, these steps did not bias or substantive analyses involving measures of adversity. To transparently show when we accessed which data, we created an open science workflow that would automate this process. The main aim of this workflow was to create a transparent log of every major milestone of the project, such as accessing new data, submitting preregistrations, and finalizing analyses.

The main ingredient of this workflow is a set of custom functions that we created for reading in data files (See Figure S1). These are wrappers for the read functions in the *readr* package. Whenever one of these functions (e.g., *read_csv*) was called, it went through a couple of internal processes. First, the specified data file would be read into R (but not yet accessible to us in the global environment). This could be a single file, or a list of individual data files that would first be combined into a single dataframe. Second, any specified manipulations would be applied to the data. This could be selecting specific variables, filtering specific rows, or randomly shuffling values (e.g., participant IDs). Third, An MD5 hash of the final R object would be generated using the *digest* package. An MD5 hash is a unique, 32-digit string that maps directly onto the content of the R object. The same R object will always generate the same MD5 hash, but as soon as anything changes (e.g., a variable is added, a value is rounded), the MD5 hash changes. Fourth, this MD5 hash would be compared to previously generated hashes.

If the newly generated MD5 hash was not recognized, this triggered an automatic commit to GitHub. At this point, the user gets the choice to abort the process or to continue. Aborting

would terminate the process without importing the data. If opting to continue, the user could supply an informative message (e.g., “accessed Flanker data”), which would be added to the Git commit. The Git commit message stored other relevant meta-data as well, such as the object hash and the code used to read and manipulate the data. Committing and pushing to Git was handled using the *gert* package.

Thus, any accessing of raw data was automatically tracked via GitHub. Using this same approach, we also logged other major milestones, such as submitting preregistrations and finalizing analyses.

An automatically generated overview of all milestones can be found in the [Data Access History](#).

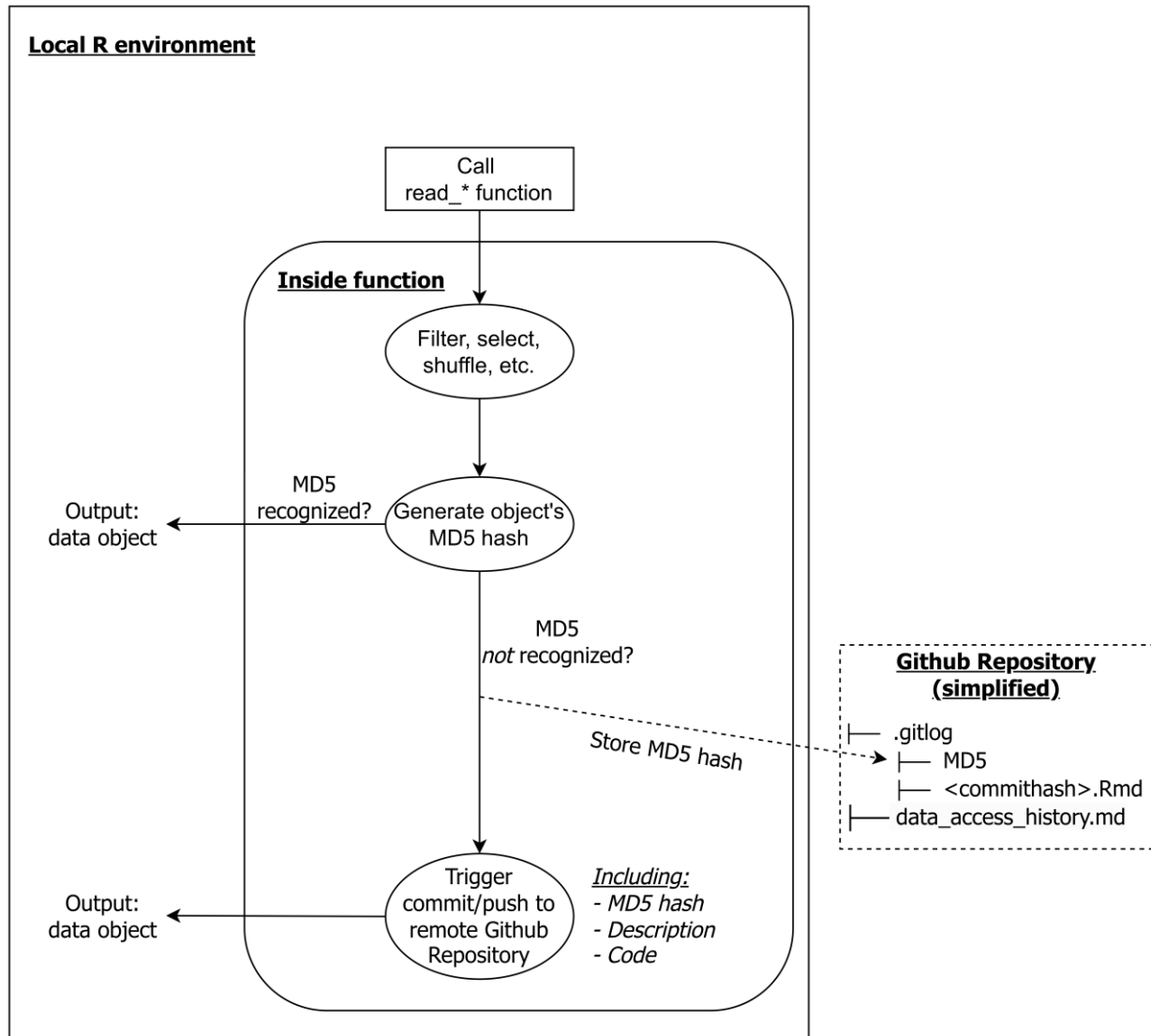


Figure S1. Graphical overview of the data access workflow using R and GitHub.

Power analysis

We conducted a power analysis through simulation using the *simulateData* function of the *lavaan* package. On each iteration, we first specified a population model (i.e., the ‘true’ model) with prespecified factor loadings and regression coefficients. Factor loadings in this model were randomly generated between 0.6 and 0.8 following a uniform distribution. Next, we simulated data sets based on the population model. Finally, we fitted a sample model (i.e., without constrained parameters) to the simulated data and extracted the beta coefficients and

corresponding p -values. We generated population models with beta coefficients of 0.06, 0.08 and 0.1, and simulated data with sample sizes ranging from 1,500 to 8,500 with steps of 1,000. Each combination of coefficients and sample sizes was repeated 500 times, for a total of 12,000 iterations.

The results are shown in Figure S2. The simulations yield power $> 90\%$ at around $N = 2,500$ for $\beta = 0.1$ and $N = 6,500$ for $\beta = 0.06$. Thus, after taking out 1,500 participants for the training set, the test set will be highly powered.

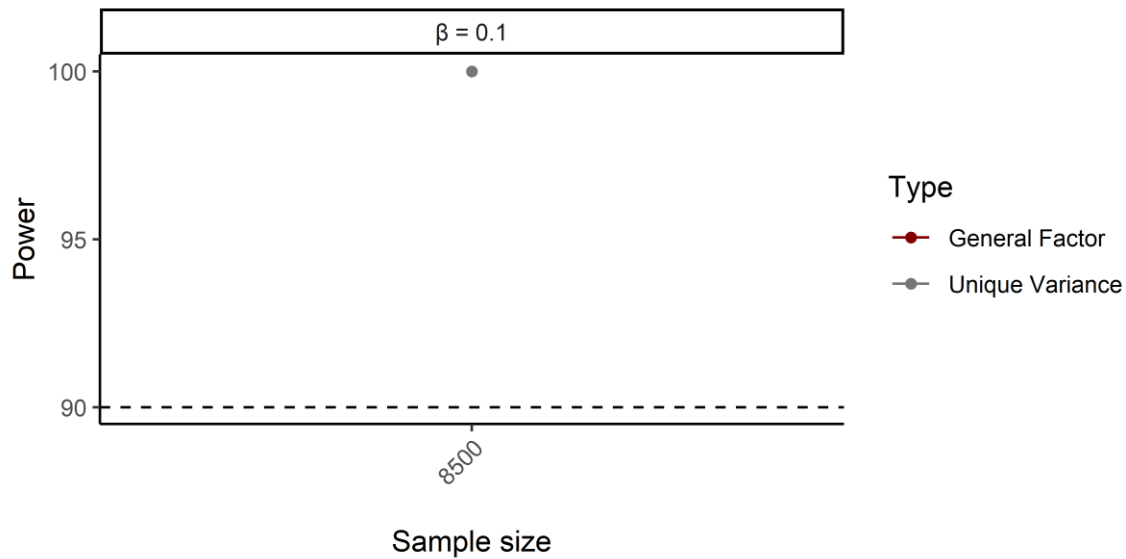


Figure S2. Results of the power simulations. The dashed line indicates 90% power.

Response Distributions of Cognitive Tasks

See Table S1 for descriptive statistics for all cognitive tasks.

Table S1. Descriptive statistics of mean RTs and accuracy for the cognitive tasks.

	RT Mean (SD)	Accuracy Mean (SD)	Accuracy Min	Accuracy Max
Processing Speed	2.24 (0.47)	96.42 (4.3)	55.17	100
Flanker	0.91 (0.33)	99.31 (3.25)	52.63	100
Mental Rotation	2.65 (0.47)	59.25 (16.81)	6.25	100
Attention Shifting	1.01 (0.35)	92.94 (6.76)	22.22	100

Overview of DDM Modeling Procedure

In theory, the hierarchical Bayesian framework allows simultaneously estimating DDM parameters, latent measurement models, and the regression paths between them in a single step (e.g., Schubert et al., 2019; Vandekerckhove, 2014). An advantage of this approach is that information regarding estimation uncertainty (e.g., of the DDM parameters) gets integrated in subsequent steps. However, this approach is very computationally expensive and might even be unfeasible with the current sample size. Therefore, we opted for a two-step estimation approach.

The hierarchical DDM models will be fit using the *runjags* package (Denwood, 2016) with JAGS code adapted from Johnson et al. (2017). The JAGS code will be adjusted in a number of ways to meet our purposes. Across all models, the starting point will be fixed to 0.5, and the boundary separation will be constrained to be the same across conditions where relevant. Each model will be fit with three Markov Chain Monte Carlo (MCMC) chains. Each chain will contain 2,000 burn-in samples and 10,000 additional samples. Of these samples, every 10th sample will be retained. Posterior samples of all three chains will be combined, resulting in a posterior sample of 3,000 samples. If a model does not converge properly with these settings, we will increase the amount of samples drawn stepwise up to 100,000.

Model convergence will be assessed in several ways. First, we will visually inspect the traces, which should not contain any drifts or large jumps. Second, we will calculate the Gelman-Rubin convergence statistic R^{\wedge} (Gelman & Rubin (1992)), of which all values should be below 1.1. Third, we will assess whether the model provides a good fit to the participants' data. To do this, we will simulate RT and accuracy data using the estimated DDM parameters of each participant (using the same number of trials as the participant). We then fit the DDM to these

simulated data using the same model specification. We will compute overall correlations between the observed and simulated scores for RTs in the 25th, 50th and 75th percentile of the RT distribution as well as for accuracies. If the correlation is $< .80$, we will take steps to improve model fit (see next paragraph). In addition, we will also compute correlations between observed and simulated RTs and accuracy at different levels of the two adversity measures: $<1SD$, $\geq 1SD$, and $>1SD$. This will tell us whether parameter recovery is worse for specific subgroups of participants, which would require caution when interpreting the results. If correlations for specific subgroups are low but the overall correlation is $> .80$, we will still use the estimates in the analyses.

In case of overall model fit $< .80$ for a particular task, we will determine criteria to find outliers based on the following simulation procedure. First, we will simulate DDM parameters for 10,000 participants based on the overall sample parameter distributions (means, standard deviations, and the variance-covariance matrix). Second, we will generate RT and accuracy data based on this new set of simulated parameters. Third, we will fit the DDM to these RT and accuracy data and again generate RT and accuracy data from these estimated DDM parameters. Thus, this procedure yields a set of simulated RT/accuracy data and corresponding recovered RT/accuracy data. We will fit regression models predicting estimated RTs and accuracy with simulated RTs and accuracy at the 25th, 50th and 75th percentile. The 2.5% and 97.5% quantiles of the residuals will be extracted from each model and used as cut-offs for bad model fit. Participants will be excluded if any of their RTs or accuracies are larger than these cut-offs. After excluding outliers, we will fit the DDM model again and repeat model fit assessments.

Imputation of the Mental Rotation Task

During preprocessing, we discovered that the 5-second response cut-off that was used for the Mental Rotation Task led to severe truncation of the RT distribution. This is problematic because the tail of the distribution holds important information about stages of processing. Truncation of reasonably long RTs can therefore lead to biased DDM parameter estimates. The hierarchical Bayesian framework allows these missing values to be imputed based on the rest of the data, which has been shown to lead to unbiased estimates. The procedure is described in detail in the supplemental materials of Johnson et al. (2017). In short, it involves two steps. First, responses are sampled probabilistically for each missing trial based on the overall accuracy of the participant. For example, if a participant has an overall accuracy of 80%, each missing response has a probability of .80 to be assigned a 1 (i.e., correct response). Second, responses are assigned to three bins. The first bin contains incorrect (imputed) RTs slower than 5 seconds (coded as -5). The second bin contains the observed data, ranging between -5 and 5 seconds. The third bin contains correct (imputed) RTs slower than 5 seconds (coded as 5). JAGS then imputes the response times for missing trials based on these thresholds. We will compare model versions with and without imputation of missing responses. A simulation demonstrating the feasibility of this approach is described below (DDM simulation 5: Imputation of missing RTs)

DDM simulations: The effect of few trials per participant

The number of trials that is available for each of the cognitive tasks is substantially lower than is typical for DDM analyses. This is especially true for the Flanker Task (8 incongruent trials, 12 congruent trials) and the Attention Shifting Task (7 switch trials, 23 repeat trials). While each participant completed a small number of trials, the hierarchical Bayesian framework can use information from the full sample to estimate and constrain individual estimates. Here, we

report simulation studies that aimed to assess whether it would be possible to accurately recover parameter estimates. The analyses are modeled on the Flanker Task, which is the task with the lowest overall number of trials ($N = 20$). For simulations involving two conditions, we assume (as we do in the real data) that the drift rate and non-decision time differ (and are correlated) across conditions, and that the boundary separation is the same across conditions. This latter assumption reflects the fact that conditions are randomly shuffled on a trial-by-trial basis, which prohibits participants from adapting their strategy for different conditions. The starting point is fixed to the mid-point (0.5) for all simulations.

DDM simulation 1: Single condition with eight trials

First, we simulated task data for 1,500 participants with eight trials per participant. We used the first 2,000 samples as burn-in, and then took an additional 10,000 samples. Every 10th sample was discarded to reduce the size of final model object. We sampled across three chains, which were subsequently combined, for a total of 3,000 samples. The model converged normally (Figure S3). Relative parameter recovery was decent for boundary separation ($r = .76$) and non-decision time ($r = .73$), but not for drift rate ($r = .54$). However, estimates of boundary separation and non-decision time showed substantial bias (See Figure S4).

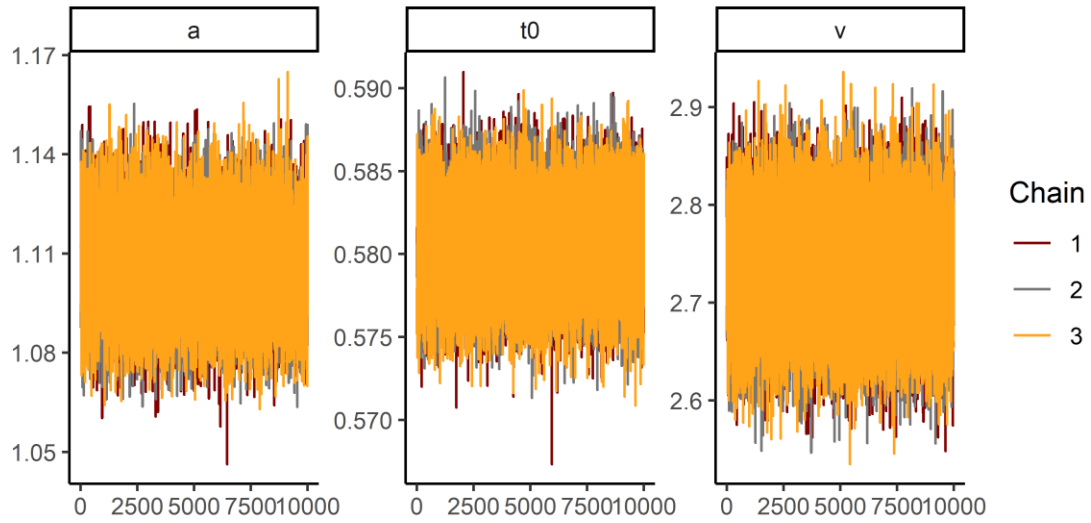


Figure S3. Convergence of the model in simulation 1. Plots should resemble a ‘fat, hairy caterpillar’.

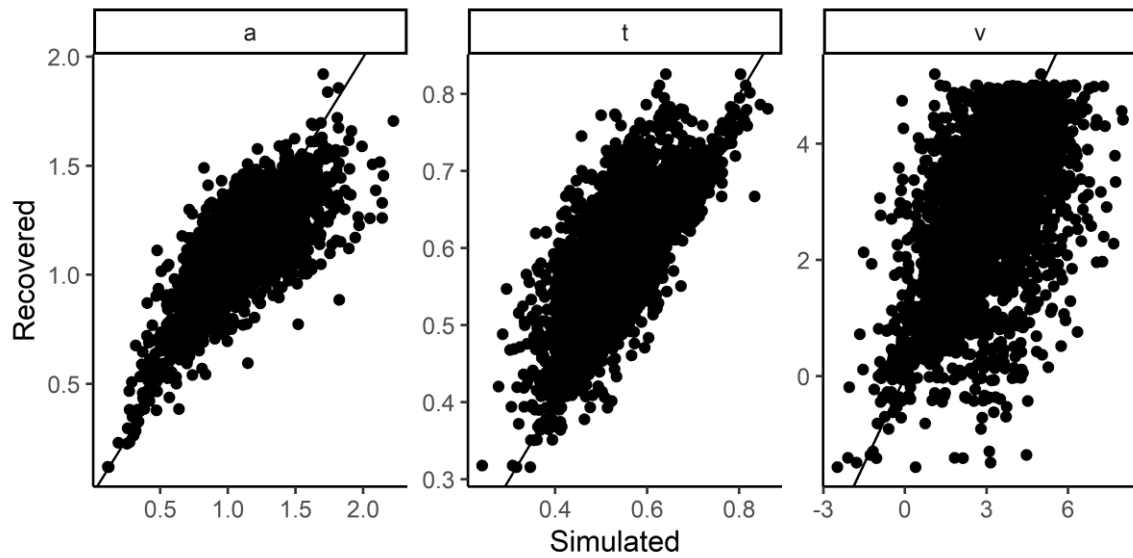


Figure S4. Parameter recovery in the case of two conditions. a = Boundary Separation; t = Non-Decision Time; v = Drift Rate.

As discussed above, the models reported in this manuscript will not be constricted to eight trials. Instead, they will be able to use the information of both conditions (e.g., congruent and incongruent for the Flanker Task), as parameters will tend to be correlated across conditions. Therefore, we ran a second simulation adding realistic condition effects.

DDM simulation 2: Two conditions; Boundary Separation fixed across conditions —

We again simulated task data for 1,500 participants. Mirroring the real Flanker task, we simulated two conditions, one with 8 trials (incongruent) and one with 12 trials (congruent). On average, drift rates were lower and non-decision times were longer for incongruent trials. Boundary separation was fixed within subjects to be equal across conditions. Non-decision times correlated on average .70 between conditions, and drift rates correlated on average .30 between conditions. These correlations were based on previous studies that we did using the Flanker Task. For more information on the specific settings, see https://github.com/stefanvermeent/abcd_ddm/scripts/0_simulations/ddm_trial_simulations.R.

As the model converged without issues in simulation 1, we tried reducing the number of samples (2,000 burn-in with an additional 2,000 samples) to save time. The model converged normally (Figure S5). Correlations between simulated and recovered parameter estimates was high, ranging between $r = .84$ for the drift rate and .95 for the non-decision time (see Figure S6).

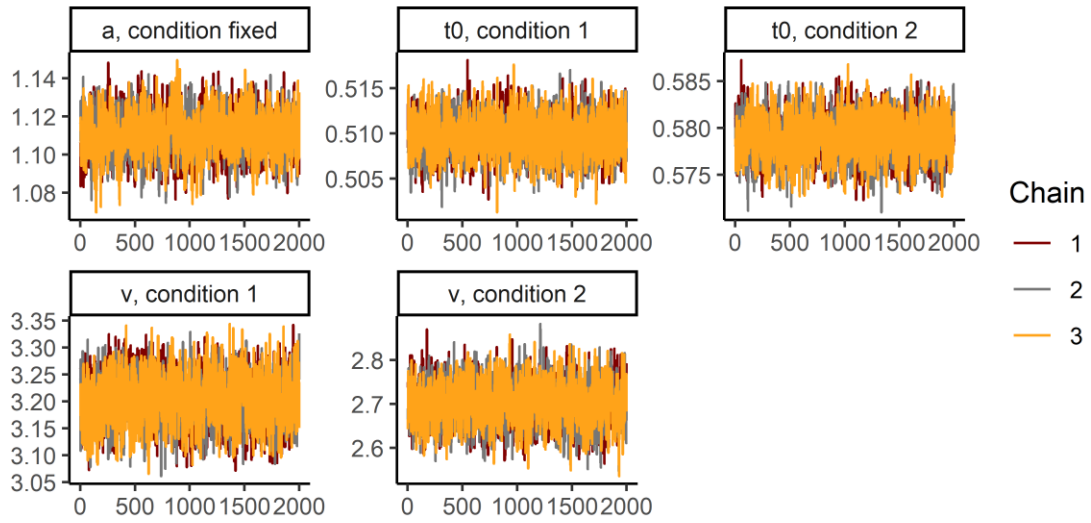


Figure S5. Convergence of the model in simulation 2. Plots should resemble a ‘fat, hairy caterpillar’.

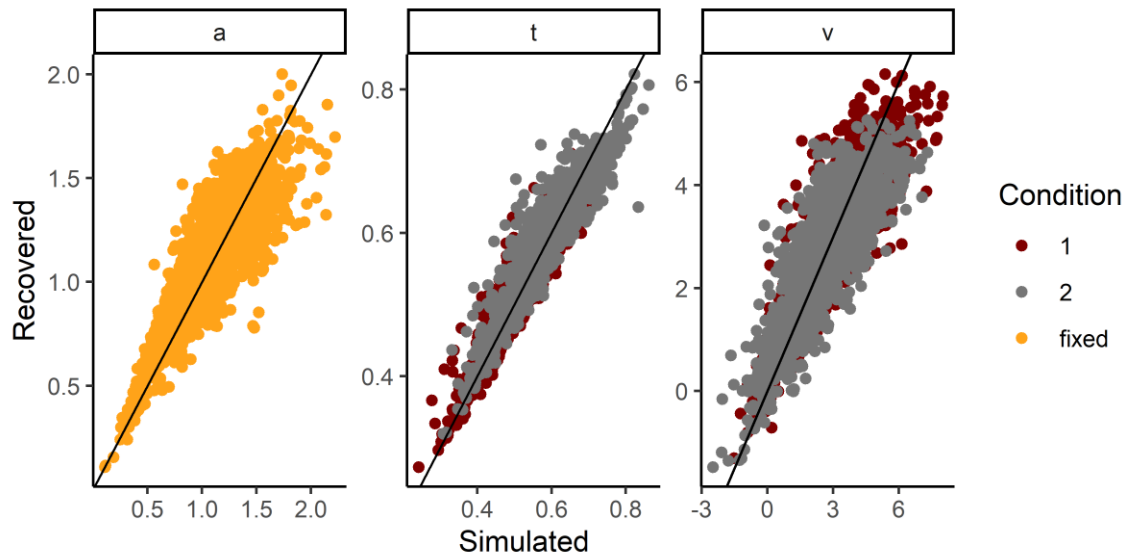


Figure S6. Parameter recovery in the case of two conditions. a = Boundary Separation; t = Non-Decision Time; v = Drift Rate

Simulation 1 and 2 involved data of 1,500 simulated subjects. However, the sample size of our real data set is roughly 10,000. Thus, in the real data there is substantially more group-level data to inform and constrain the individual parameter estimates. we ran a third simulation to investigate if—and to what extent—the parameter estimates would improve moving from 1,500 to 10,000 participants.

DDM simulation 3: Two conditions; 10,000 subjects

We simulated task data for 10,000 participants. All other simulation settings were identical to simulation 2.

The model converged normally (Figure S7). Correlations between simulated and recovered parameter estimates were high and very similar to those found in simulation 2, ranging between $r = .83$ for the drift rate and $.95$ for the non-decision time (see Figure S8). Thus, the benefit of adding more subjects is already saturated around 1,500 participants, with additional participants not improving parameter estimation.

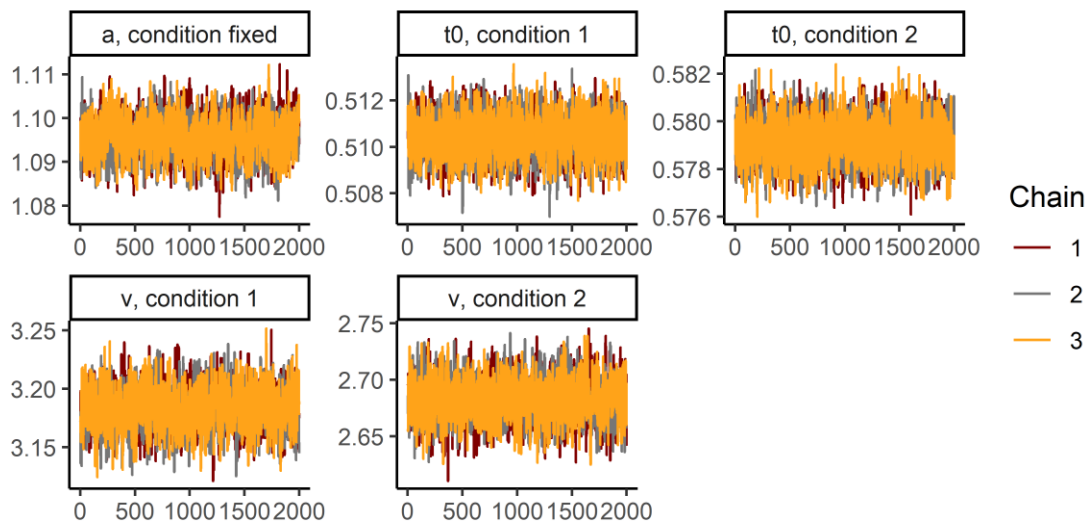


Figure S7. Convergence of the model in simulation 3. Plots should resemble a ‘fat, hairy caterpillar’.

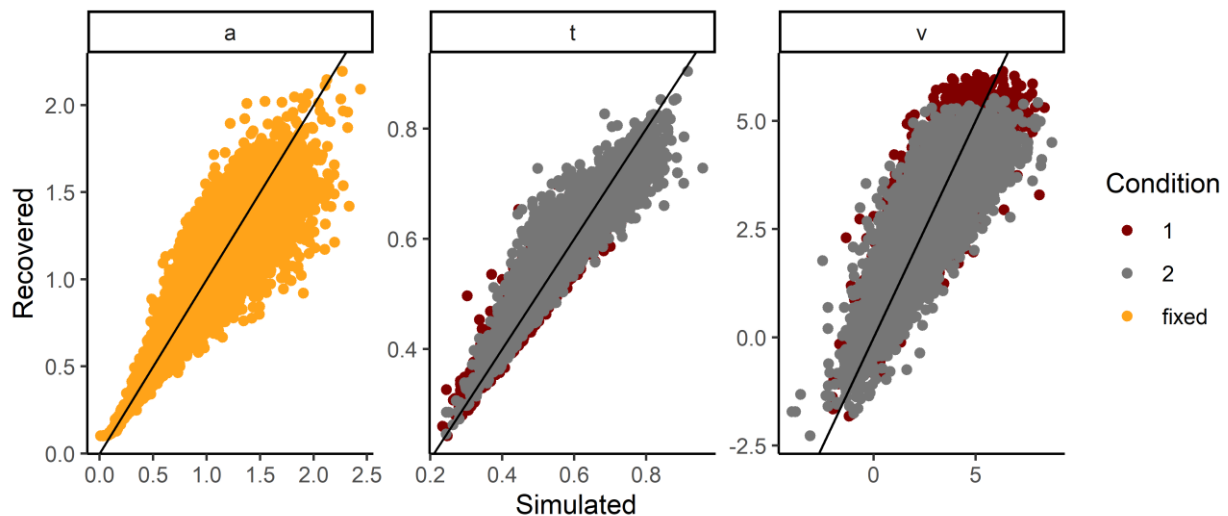


Figure S8. Parameter recovery in the case of two conditions. a = Boundary Separation; t = Non-Decision Time; v = Drift Rate

Overall, we conclude that applying hierarchical Bayesian DDM to the ABCD data is feasible.

Additional DDM simulations

DDM simulation 4: Does shrinkage bias the associations between parameter estimates and adversity?

One of the reviewers noted that the hierarchical Bayesian DDM tends to compress parameter estimates by pulling extreme values toward the group mean (a phenomenon known as shrinkage). A concern may be that this could potentially reduce the individual differences of interest, especially if these occur in the tail of the distribution (e.g., the participants with the highest levels of adversity obtaining the most extreme parameter estimates). In general, this is not the case; in contrast, shrinkage tends to pull less reliable and outlier estimates towards the group mean, which has been shown to positively affect the signal-to-noise ratio and reliability of parameter estimates in cognitive neuroscience (Dai et al., 2017; Mejia et al., 2018). To

specifically study the effects of shrinkage on the variance of DDM parameter estimates, we nevertheless ran a simulation to investigate the likelihood that shrinkage might obscure adversity-DDM parameter associations.

We simulated DDM parameters for 1,500 participants. Participants' adversity scores followed a log-normal distribution ($\text{mean}_{\log} = 0$, $\text{sd}_{\log} = 0.3$) to approximate the skew in the right tail typically found in adversity scores. Drift rates were simulated based on a standardized association of $\beta = 0.1$ with the adversity score. Thus, higher levels of adversity tended to be associated with higher drift rates. Based on the simulated DDM parameters, we simulated 20 trials (RTs and accuracy) per participant, which were then used as input to the DDM model. We used the first 2,000 samples as burn-in, and then took an additional 2,000 samples. We sampled across three chains, which were subsequently combined, for a total of 6,000 samples.

All parameters were recovered with high correlations ranging between 0.84 and 0.97. Figure S9 shows signs of shrinkage, especially in the right tail of the drift rate distribution. However, the difference in standard deviations was minimal ($SD_{\text{simulated}} = 1.52$; $SD_{\text{recovered}} = 1.46$).

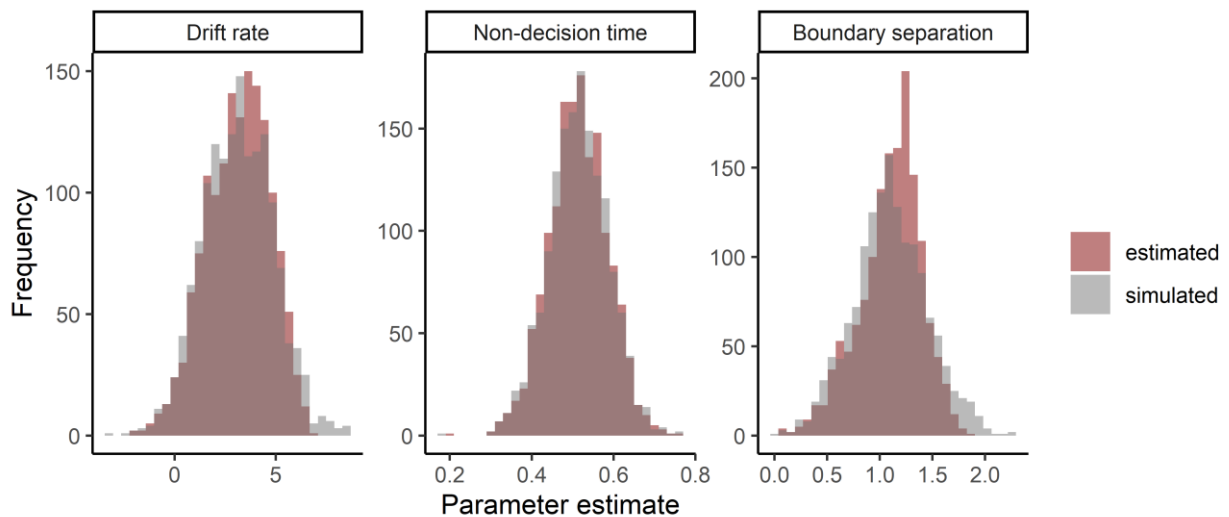


Figure S9. Histograms of simulated and recovered parameter estimates. a = Boundary Separation; t = Non-Decision Time; v = Drift Rate.

Next, we calculated the deviations between each simulated and recovered parameter estimate and plotted this against the adversity scores (See Figure S10). None of the associations were statistically significant (all $ps > .05$ for linear and quadratic effects).

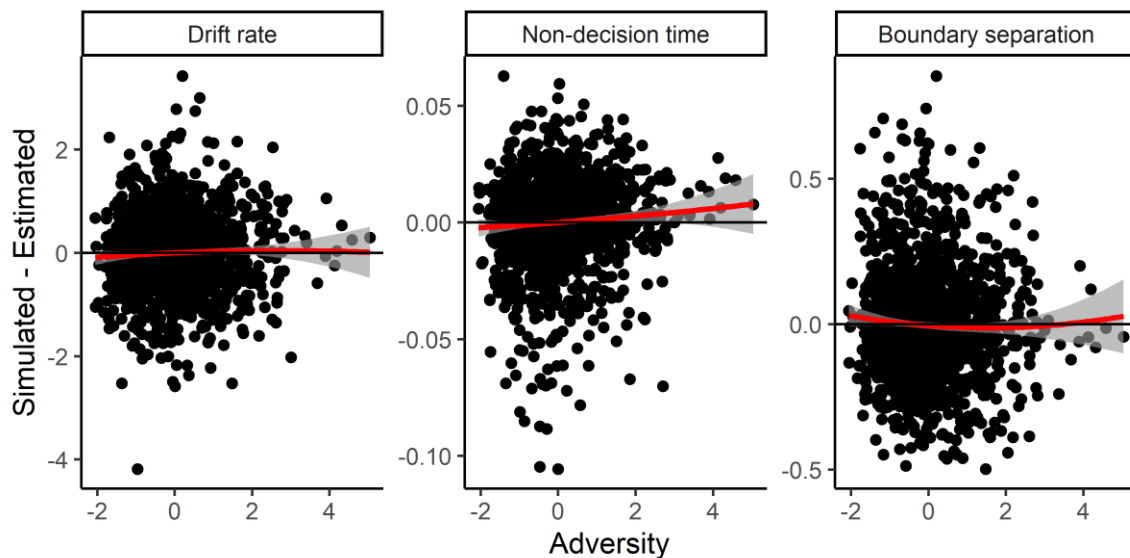


Figure S10. Deviation between simulated and recovered parameter estimates as a function of adversity. The regression lines show quadratic effects. a = Boundary Separation; t = Non-Decision Time; v = Drift Rate.

Finally, we fitted a linear mixed model predicting drift rates estimates as a function of adversity, dataset (simulated vs. recovered; dummy-coded with simulated as the reference category), and the adversity x dataset interaction to assess whether the difference between simulated and recovered drift rates would be different at low, average, and high levels of adversity. We did not find a significant adversity x dataset interaction, $b = -0.03$, $p = .163$. As

Figure 11 illustrates, there seemed to be small shrinkage effects at the low and high levels of adversity. However, none of these simple slope effects were statistically significant. Taken together, we conclude that DDM recovery at higher levels of adversity was not less precise compared to lower levels of adversity.

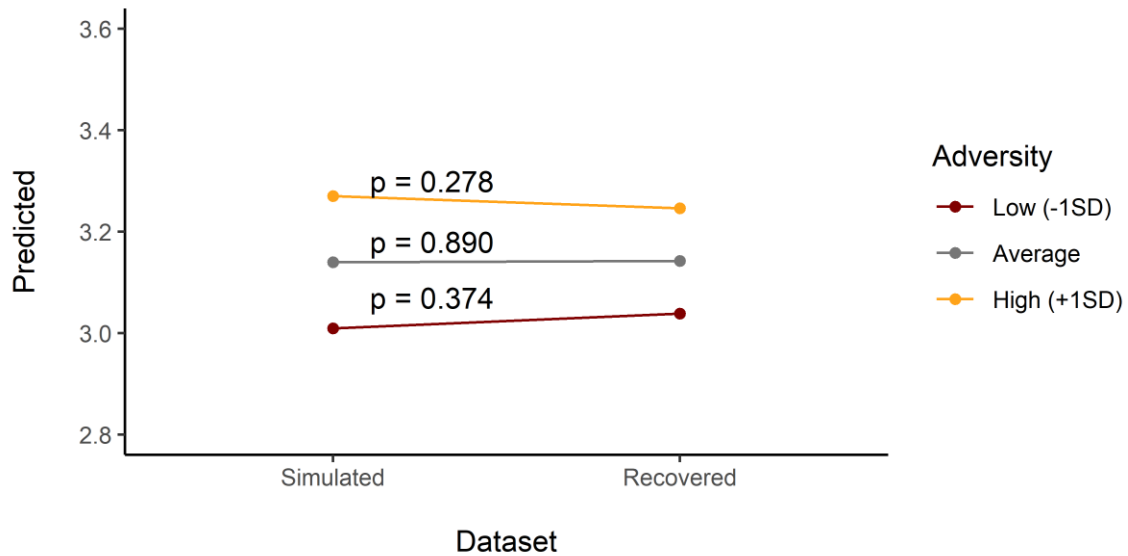


Figure S11. Simple slopes of the difference between simulated and estimated drift rates at different levels of adversity.

DDM simulation 5: Imputation of missing RTs

To demonstrate the feasibility of the imputation approach for the Mental Rotation Task, we ran a simulation based on 1500 participants in which RT and accuracy data were generated modeled on the real Mental Rotation Task data (RT: $M_{\text{real}} = 2.65$, $M_{\text{sim}} = 2.76$; Accuracy: $M_{\text{real}} = 59.25\%$, $M_{\text{sim}} = 67.23\%$; RTs above 5 s cut-off: $M_{\text{real}} = 10.04\%$, $M_{\text{sim}} = 8.18\%$). We fitted two DDM models: one that was fit to the complete data (including RTs > 5 s) and one that was fit to data in which all RTs > 5 s were set to missing. In the latter case, missing RTs were imputed as

described above. All other model fit settings were identical to simulations 2-4. Correlations between DDM parameters based on the complete data and imputed data were near perfect, $r = 1$ for drift rate, $r = 0.996$ for non-decision time, and $r = 0.993$ for boundary separation.

DDM Model Fit Assessments

Parameter recovery

A fit overview for all DDM models is presented in Table S2-S5. As described in the main article, we deviated from our preregistered fit procedure by simulating 100 trials per task instead of the same amount as the observed trials. These results are presented in Table S2. We also present the results using the original, preregistered approach in Table S3.

Table S2. Simulation-based model fit assessment comparing observed and predicted data using 100 simulated trials (accuracy, 25th, 50th, 75th percentile).

Task	Condition	25th Percentile	50th Percentile	75th Percentile	Accuracy	R ²
Flanker - Model 1	congruent	0.96	0.96	0.95	0.53	1.008
Flanker - Model 1	incongruent	0.94	0.95	0.94	0.93	1.008
Flanker - Model 2		0.96	0.96	0.96	0.90	1.011
Mental Rotation - Model 1		0.90	0.88	0.84	0.94	1.010
Mental Rotation - Model 2		0.87	0.88	0.86	0.96	1.010
Attention Shifting - Model 1	repeat	0.80	0.82	0.80	-0.00	1.009
Attention Shifting - Model 1	switch	0.82	0.82	0.83	0.46	1.009
Attention Shifting - Model 2		0.94	0.95	0.95	0.88	1.008
Processing Speed - Model 1		0.92	0.94	0.93	0.81	1.013
Processing Speed - Model 2		0.94	0.94	0.93	0.80	1.011

Note: The models that were selected for inclusion in the primary analyses are printed in bold.

Table S3. Simulation-based model fit assessment comparing observed and predicted data using the same number of observed and simulated trials (accuracy, 25th, 50th, 75th percentile).

Task	Condition	25th Percentile	50th Percentile	75th Percentile	Accuracy	R ²
Flanker - Model 1	congruent	0.88	0.88	0.87	0.30	1.008
Flanker - Model 1	incongruent	0.88	0.89	0.88	0.88	1.008
Flanker - Model 2		0.91	0.92	0.93	0.79	1.011
Mental Rotation - Model 1		0.84	0.80	0.76	0.87	1.010
Attention Shifting - Model 1	repeat	0.72	0.74	0.73	-0.01	1.009
Attention Shifting - Model 1	switch	0.69	0.67	0.68	0.26	1.009
Attention Shifting - Model 2		0.91	0.91	0.90	0.73	1.008
Processing Speed - Model 1		0.88	0.89	0.88	0.66	1.013
Processing Speed - Model 2		0.90	0.89	0.88	0.65	1.011

Note: The models that were selected for inclusion in the primary analyses are printed in bold.

As planned, we explored whether model fit would be relatively worse at different levels of the two measures of adversity. Table S4 and S5 present correlations between observed and predicted RTs and accuracy at different levels of adversity. Model fit was high for all tasks across all levels of adversity, and there were no indications for any meaningful differences.

Table S4. Simulation-based model fit assessment at different levels of material deprivation comparing observed and predicted data using 100 simulated trials (accuracy, 25th, 50th, 75th percentile).

Task	Material deprivation	25th Percentile	50th Percentile	75th Percentile	Accuracy
Flanker - Model 2	< -1SD	0.97	0.97	0.96	0.82
Flanker - Model 2	> 1SD	0.95	0.95	0.96	0.93
Flanker - Model 2	≥1*SD*≤	0.96	0.96	0.96	0.87
Mental Rotation - Model 1	< -1SD	0.88	0.86	0.82	0.94
Mental Rotation - Model 1	≥1*SD*≤	0.89	0.88	0.84	0.94
Mental Rotation - Model 1	> 1SD	0.92	0.89	0.85	0.93
Attention Shifting - Model 2	< -1SD	0.94	0.95	0.95	0.85
Attention Shifting - Model 2	> 1SD	0.93	0.94	0.94	0.90
Attention Shifting - Model 2	≥1*SD*≤	0.94	0.95	0.95	0.86
Processing Speed - Model 2	< -1SD	0.95	0.96	0.94	0.82
Processing Speed - Model 2	≥1*SD*≤	0.94	0.94	0.93	0.79
Processing Speed - Model 2	> 1SD	0.93	0.93	0.93	0.81

Table S5. Simulation-based model fit assessment at different levels of household threat comparing observed and predicted data using 100 simulated trials (accuracy, 25th, 50th, 75th percentile).

Task	Household threat	25th Percentile	50th Percentile	75th Percentile	Accuracy
Flanker - Model 2	< -1SD	0.96	0.96	0.96	0.83
Flanker - Model 2	> 1SD	0.96	0.96	0.95	0.92
Flanker - Model 2	$\geq 1*SD \leq$	0.96	0.96	0.96	0.88
Mental Rotation - Model 1	< -1SD	0.88	0.87	0.82	0.94
Mental Rotation - Model 1	$\geq 1*SD \leq$	0.90	0.88	0.84	0.94
Mental Rotation - Model 1	> 1SD	0.90	0.88	0.85	0.94
Attention Shifting - Model 2	< -1SD	0.95	0.95	0.95	0.84
Attention Shifting - Model 2	> 1SD	0.94	0.94	0.94	0.86
Attention Shifting - Model 2	$\geq 1*SD \leq$	0.94	0.95	0.95	0.88
Processing Speed - Model 2	< -1SD	0.94	0.94	0.94	0.79
Processing Speed - Model 2	$\geq 1*SD \leq$	0.94	0.94	0.93	0.80
Processing Speed - Model 2	> 1SD	0.93	0.93	0.93	0.81

For the Processing Speed Task, we found a high degree of Kurtosis in the left-hand tail of the non-decision time distribution. This tail consisted of participants with the lowest RTs (between 0.3s and ~1s). Although overall accuracy on the Processing Speed Task was very high (96.41%), overall accuracy for RTs < 1s was below chance, with accuracy increasing above change starting at 1s. Therefore, we decided to remove RTs < 1s (0.1% of trials) and refit the model, which solved the kurtosis in non-decision times. Recovery of RTs was above the cut-off of .80 for each quantile. Thus, we selected this model for the main analyses.

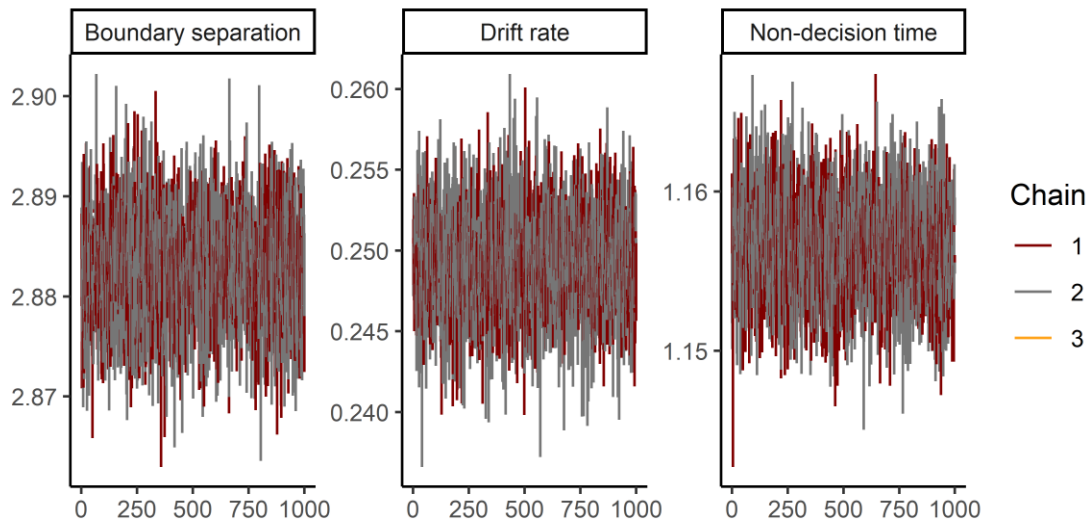
Model convergence

Figure S12. Convergence of the final model for the Mental Rotation Task. Plots should resemble a ‘fat, hairy caterpillar’.

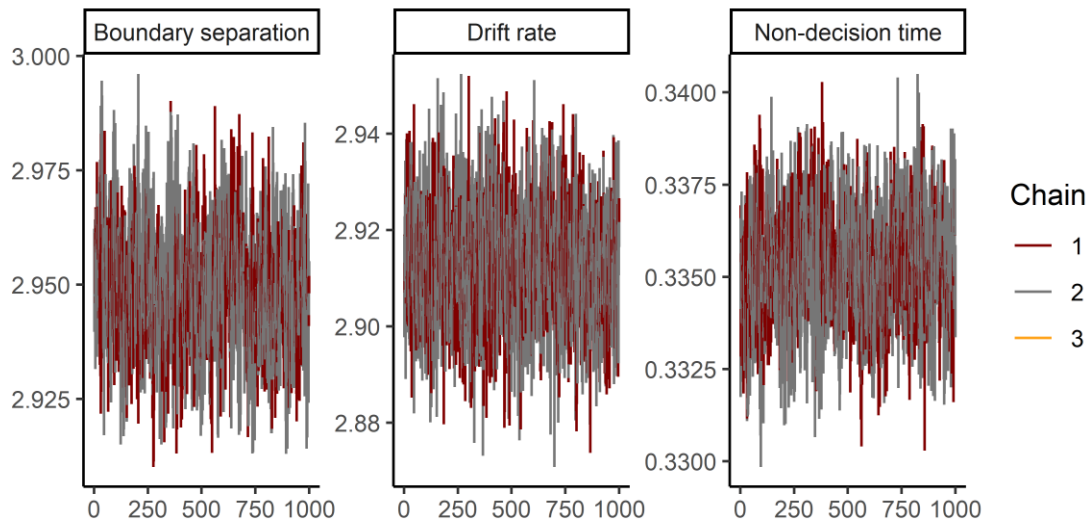


Figure S13. Convergence of the final model for the Inhibition Task. Plots should resemble a ‘fat, hairy caterpillar’.

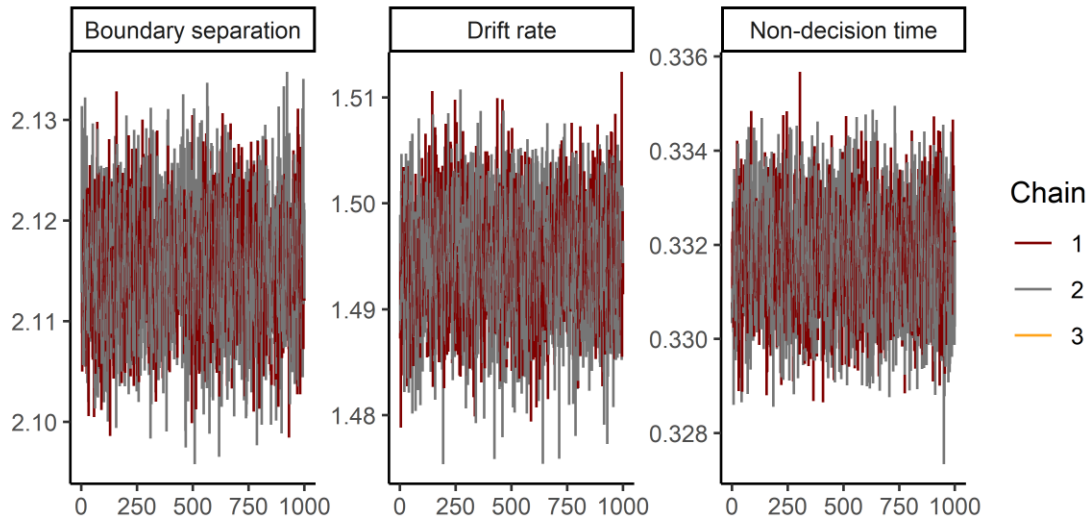


Figure S14. Convergence of the final model for the Attention Shifting Task. Plots should resemble a 'fat, hairy caterpillar'.

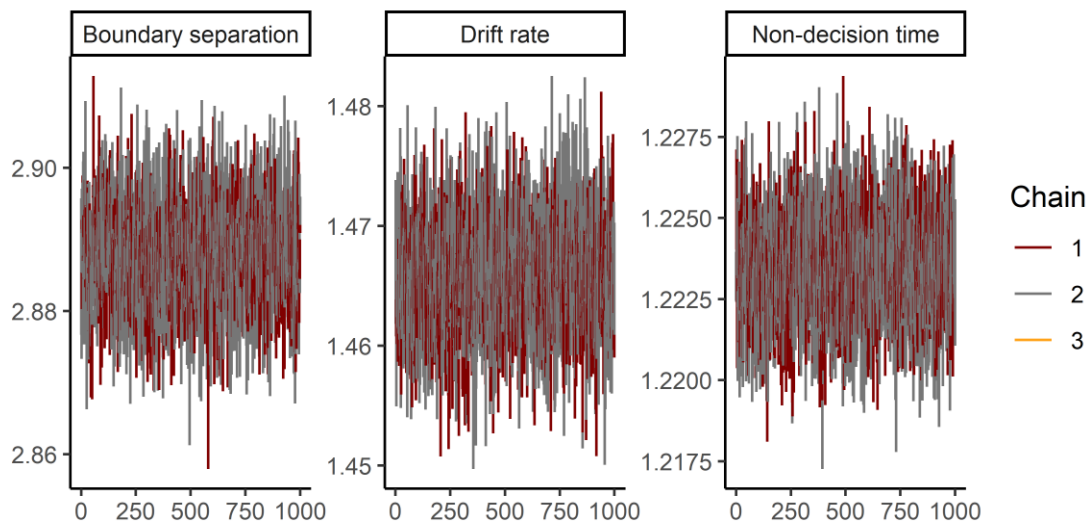


Figure S15. Convergence of the final model for the Processing Speed Task. Plots should resemble a 'fat, hairy caterpillar'.

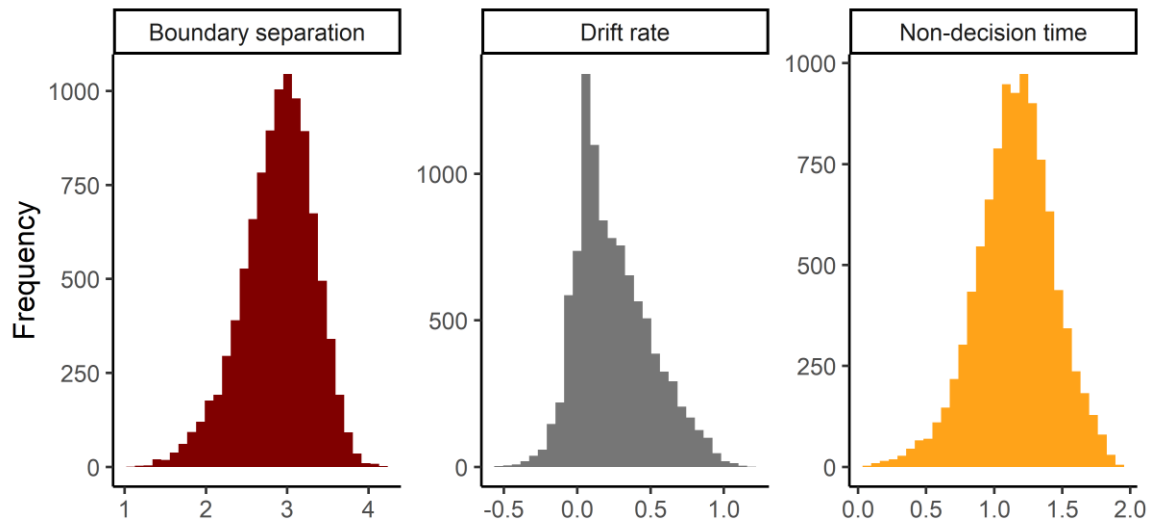
Parameter distributions

Figure S16. Parameter distributions in the final model of the Mental Rotation Task.

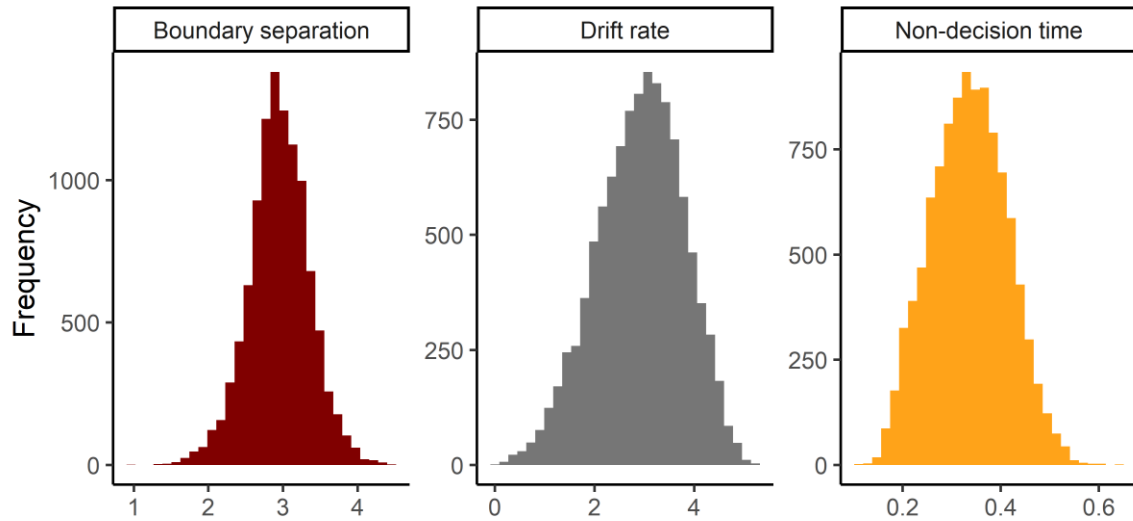


Figure S17. Parameter distributions in the final model of the Inhibition Task.

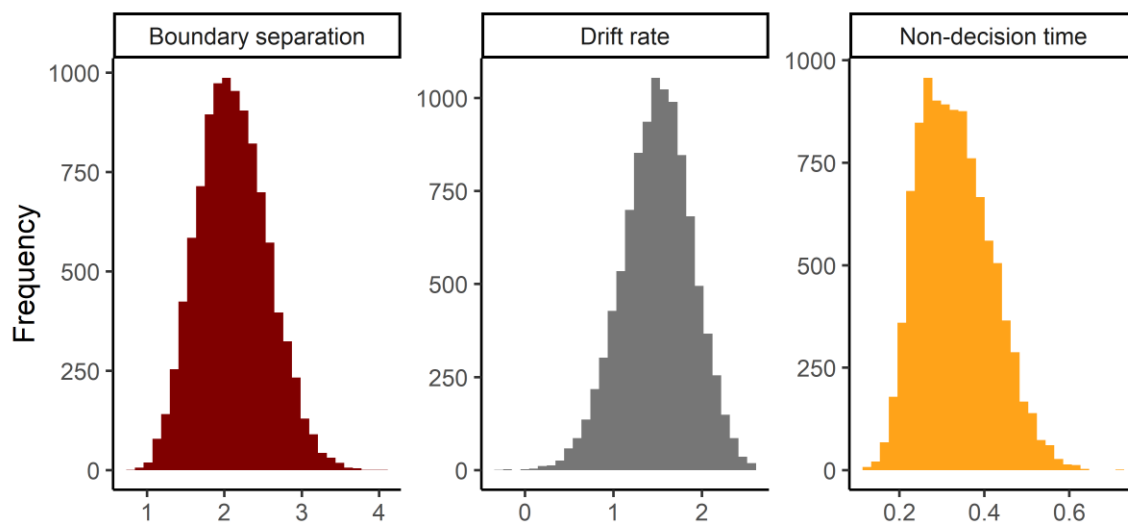


Figure S18. Parameter distributions in the final model of the Attention Shifting Task.

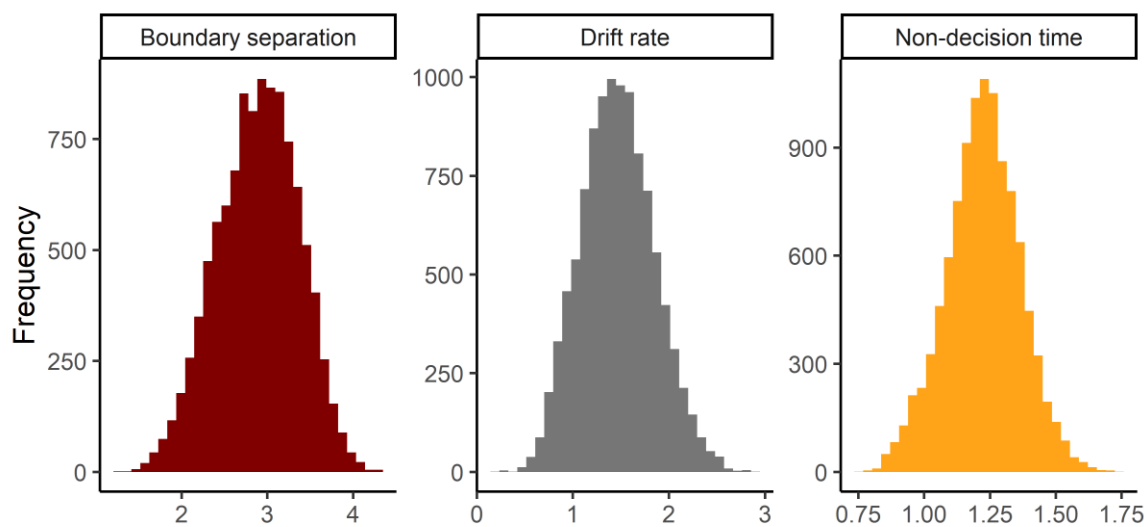


Figure S19. Parameter distributions in the final model of the Processing Speed Task.

SEM Fit**Table S6.** Factor loadings and unstandardized residual variances in the test set.

	Estimate (unstandardized)	SE	Z	p	Estimate standardized
Factor loadings					
Task-general drift rate					
Processing Speed Task	1.00	0.00			0.52
Attention Shifting Task	1.19	0.04	29.43	0.000	0.63
Mental Rotation Task	0.51	0.03	18.37	0.000	0.27
Inhibition Task	1.22	0.04	29.76	0.000	0.65
Task-general boundary separation					
Processing Speed Task	1.00	0.00			0.55
Attention Shifting Task	1.44	0.04	38.19	0.000	0.80
Mental Rotation Task	0.28	0.02	11.97	0.000	0.15
Inhibition Task	1.14	0.03	37.63	0.000	0.63
Task-general non-decision time					
Processing Speed Task	1.00	0.00			0.45
Attention Shifting Task	1.46	0.05	27.28	0.000	0.66
Mental Rotation Task	0.67	0.03	19.49	0.000	0.30
Inhibition Task	1.53	0.05	29.78	0.000	0.70
Residual variances					
Task-specific drift rate					
Inhibition Task	0.55	0.01	43.01	0.000	
Attention Shifting Task	0.52	0.01	42.61	0.000	
Mental Rotation Task	0.82	0.01	64.81	0.000	
Processing Speed Task	0.71	0.01	55.58	0.000	
Task-specific boundary separation					
Inhibition Task	0.61	0.01	53.53	0.000	
Attention Shifting Task	0.39	0.01	31.86	0.000	
Mental Rotation Task	0.95	0.01	66.89	0.000	
Processing Speed Task	0.69	0.01	58.55	0.000	
Task-specific non-decision time					
Inhibition Task	0.56	0.01	44.13	0.000	
Attention Shifting Task	0.61	0.01	44.01	0.000	
Mental Rotation Task	0.86	0.01	64.90	0.000	
Processing Speed Task	0.78	0.01	60.72	0.000	

Table S7. Correlations between latent task-general and task-specific factors in the test set.

	Correlation
Task-general	
Drift rate - Boundary separation	-0.575***
Drift rate - Non-decision time	-0.046*
Boundary separation - Non-decision time	0.710***
Task-specific Inhibition Task	
Drift rate - Boundary separation	-0.098***
Drift rate - Non-decision time	0.019
Boundary separation - Non-decision time	0.340***
Task-specific Attention Shifting Task	
Drift rate - Boundary separation	-0.106***
Drift rate - Non-decision time	0.030*
Boundary separation - Non-decision time	-0.228***
Task-specific Mental Rotation Task	
Drift rate - Boundary separation	0.305***
Drift rate - Non-decision time	0.230***
Boundary separation - Non-decision time	0.102***
Task-specific Processing Speed Task	
Drift rate - Boundary separation	-0.125***
Drift rate - Non-decision time	0.052***
Boundary separation - Non-decision time	-0.097***

References

- Dai, T., Guo, Y., & Alzheimer's Disease Neuroimaging Initiative. (2017). Predicting individual brain functional connectivity using a Bayesian hierarchical model. *NeuroImage*, 147, 772–787. <https://doi.org/https://doi.org/10.1016/j.neuroimage.2016.11.048>
- Denwood, M. J. (2016). Runjags: An R package providing interface utilities, model templates, parallel computing methods and additional distributions for MCMC models in JAGS. *Journal of Statistical Software*, 71, 1–25. <https://doi.org/10.18637/jss.v071.i09>
- Gelman, A., & Rubin, D. B. (1992). Inference from iterative simulation using multiple sequences. *Statistical Science*, 457–472. <https://doi.org/10.1214/ss/1177011136>
- Johnson, D. J., Hopwood, C. J., Cesario, J., & Pleskac, T. J. (2017). Advancing research on cognitive processes in social and personality psychology: A hierarchical Drift Diffusion Model primer. *Social Psychological and Personality Science*, 8(4), 413–423. <https://doi.org/10.1177/1948550617703174>
- Mejia, A. F., Nebel, M. B., Barber, A. D., Choe, A. S., Pekar, J. J., Caffo, B. S., & Lindquist, M. A. (2018). Improved estimation of subject-level functional connectivity using full and partial correlation with empirical Bayes shrinkage. *NeuroImage*, 172, 478–491. <https://doi.org/https://doi.org/10.1016/j.neuroimage.2018.01.029>
- Schubert, A.-L., Nunez, M. D., Hagemann, D., & Vandekerckhove, J. (2019). Individual differences in cortical processing speed predict cognitive abilities: A model-based cognitive neuroscience account. *Computational Brain & Behavior*, 2(2), 64–84.
- Vandekerckhove, J. (2014). A cognitive latent variable model for the simultaneous analysis of behavioral and personality data. *Journal of Mathematical Psychology*, 60, 58–71. <https://doi.org/10.1016/j.jmp.2014.06.004>