Supplementary materials to 'Effect of spatially filtered primes on emotion categorization with an interference context in autism and typically developing'

Adeline Lacroix[1], Ladislas Nalborczyk[2], Frédéric Dutheil[3], Klara Kovarski[4,5], Sylvie Chokron[4,5], Marta Garrido[6], Marie Gomot[7], & Martial Mermillod[1]

[1] LPNC, Univ. Grenoble Alpes, Univ. Savoie Mont Blanc, CNRS, 38000, Grenoble, France
[2] GIPSA-lab, CNRS, Univ. Grenoble Alpes, 38000, Grenoble, France.
[3] Université Clermont Auvergne, CNRS, LaPSCo, CHU Clermont-Ferrand, WittyFit, F-63000 Clermont-Ferrand, France
[4] Hôpital Fondation Ophtalmologique A. de Rothschild, Paris, France.
[5] Université de Paris, INCC UMR 8002, CNRS, F-75006 Paris, France
[6] Cognitive Neuroscience and Computational Psychiatry Lab, Melbourne School of Psychological Sciences, The University of Melbourne, Australia
[7] UMR 1253 iBrain, Université de Tours, Inserm, Tours, France

Contents

Correspondence concerning this article should be addressed to Adeline Lacroix, Laboratoire de Psychologie et NeuroCognition, Univ. Grenoble Alpes, 1251 avenue centrale, Grenoble, France. E-mail: adeline.lacroix@univ-grenoble-alpes.fr

1 Model comparison for the response time analysis

The first analysis was conducted on our data without:

- participants with accuracy < 50 % (1 autistic participant)

- trials with RT < 300 ms (1,50 %)

1.1 Inclusion of different predictors

We first compared the three following models of increasing complexity and we kept the best one (i.e., the model with the lowest LOOIC - Leave-one-out cross validation information criterion - and the highest weight):

- BMgcp : RT ~ 1 + GroupC * PrimeC * CongC + (1 + CongC * PrimeC || ID) + (1 + GroupC * PrimeC * CongC || Target)

- BMgcpe : RT ~ 1 + GroupC * PrimeC * CongC * EmoC + (1 + CongC * PrimeC *EmoC || ID) + (1 + GroupC * PrimeC * CongC * EmoC || Target)

- BMgcpes : RT ~ 1 + GroupC * PrimeC * CongC * EmoC * SexC + (1 + CongC * PrimeC * EmoC || ID) + (1 + GroupC * PrimeC * CongC * EmoC * SexC || Target)

- BMgcpesP : RT ~ 1 + GroupC * CongC * PrimeC * EmoC + SexC + SexC:GroupC + (1 + CongC * PrimeC * EmoC || ID) + (1 + GroupC * CongC * PrimeC *EmoC + SexC + SexC:GroupC|| Target)

Table 1

Expected log pointise predictive density (ELPD) and LOOIC for each BLMM on RT.

model	elpd_diff	se_diff	elpd_loo	se_elpd_loo	p_loo	se_p_loo	looic	se_looic	Model_Weights
BMgcpe	0.000	0.000	-92,727.743	122.793	282.308	8.677	185,455.485	245.587	0.745
BMgcpesP	-1.408	0.755	-92,729.151	122.819	288.954	8.851	185,458.301	245.638	0.182
BMgcpes	-2.323	4.451	-92,730.066	123.017	326.123	9.920	185,460.131	246.034	0.073
BMgcp	-180.740	21.709	-92,908.483	122.026	160.269	5.267	185,816.965	244.051	0.000

As reported on *Table 1*, `BMgcpe` appeared to be the best model compared with models without emotion or with sex (full model or parsimonious model). Hence, we kept this model and then added covariates to this model.

1.2 Model with covariates

As preregistered, we added some covariates in the model and again, we kept the best one after model comparison. We included either PIQ or FSIQ and AQ as our two groups differed on these variables.

We compared the following models:

- BMgcpe : RT ~ 1 + GroupC * PrimeC * CongC * EmoC + (1 + CongC * PrimeC *EmoC || ID) + (1 + GroupC * PrimeC * CongC * EmoC || Target)*

- BMf : RT ~ 1 + GroupC * CongC * PrimeC * EmoC + FSIQc + (1 + CongC * PrimeC * EmoC || ID) + (1 + GroupC * CongC * PrimeC * EmoC || Target)

- BMa : RT ~ 1 + GroupC * CongC * PrimeC * EmoC + AQc + (1 + CongC * PrimeC * EmoC || ID) + (1 + GroupC * CongC * PrimeC *EmoC || Target)

- BMp : RT ~ 1 + GroupC * CongC * PrimeC * EmoC + PIQc + (1 + CongC * PrimeC * EmoC || ID) + (1 + GroupC * CongC * PrimeC *EmoC || Target)

- BMfa : RT ~ 1 + GroupC * CongC * PrimeC * EmoC + FSIQc + AQc + (1 + CongC * PrimeC * EmoC || ID) + (1 + GroupC * CongC * PrimeC *EmoC || Target)

- BMpa : RT ~ 1 + GroupC * CongC * PrimeC * EmoC + PIQc + AQc + (1 + CongC * PrimeC * EmoC || ID) + (1 + GroupC * CongC * PrimeC *EmoC || Target)

Table 2

Expected log pointise predictive density (ELPD) and LOOIC for each of the four BLMMs on RT, with covariates

model	elpd_diff	se_diff	elpd_waic	se_elpd_waic	p_waic	se_p_waic	waic	se_waic	Model_Weights
BMgcpe	0.000	0.000	-92,726.667	122.773	281.232	8.621	185,453.334	245.545	0.292
BMpa	-0.011	0.268	-92,726.678	122.772	281.177	8.577	185,453.356	245.543	0.278
BMf	-0.607	0.348	-92,727.274	122.797	281.521	8.645	185,454.548	245.594	0.161
BMa	-0.836	0.251	-92,727.503	122.775	281.669	8.625	185,455.006	245.549	0.126
BMfa	-1.238	0.305	-92,727.905	122.795	281.247	8.604	185,455.810	245.590	0.078
BMp	-1.430	0.284	-92,728.097	122.804	282.796	8.688	185,456.194	245.608	0.064

The model without any covariates appeared to be the best model (see *Table 2*), despite the model with AQ and PIQ seems to be almost equally good. The outcome of these two models are very similar and we kept the more parsimonious (note that there is no evidence of an effect of AQ and PIQ on our data when looking at the output of the model with covariates). We checked the quality of this model with posterior predictive checking, which seemed to be good (*Figure 3* of the main paper). The results of this BLMM are summarized in *Table 3* and are described in the paper.

Table 3

Estimates and BF01 from the BLMM on RT.

Predictors	Estimate	SE	Lower	Upper	Rhat	BF01
Intercept	696.93	13.69	670.39	724.09	1.00	NA
Group	66.78	24.40	16.93	113.38	1.00	0.069
Prime	-12.91	2.97	-18.70	-6.93	1.00	1.277×10^{-4}
Congruency	-14.78	2.88	-20.64	-9.17	1.00	1.481×10^{-4}
Emotion	18.90	6.73	5.59	31.99	1.00	0.194
Group x Prime	-1.30	5.84	-12.74	10.12	1.00	8.487
Group x Congruency	-2.42	5.43	-13.10	8.15	1.00	8.386
Prime x Congruency	4.53	4.54	-4.36	13.56	1.00	6.479
Group x Emotion	-8.04	9.48	-26.65	10.53	1.00	3.732
Prime x Emotion	3.70	4.59	-5.32	12.66	1.00	8.184
Congruency x Emotion	-4.13	4.60	-13.22	4.80	1.00	7.517
Group x Prime x Congruency	2.28	8.63	-14.63	19.04	1.00	5.494
Group x Prime x Emotion	21.19	9.23	3.08	39.21	1.00	0.403
Group x Congruency x Emotion	6.08	9.11	-11.87	23.76	1.00	4.594
Prime x Congruency x Emotion	-8.07	9.15	-26.34	9.98	1.00	3.671
Group x Prime x Congruency x Emotion	7.98	17.00	-25.18	41.78	1.00	2.666

Note. For each predictor, the first two columns represent the estimated most probable value for the slope and its standard error (SE). The 'Lower' and 'Upper' are the bounds of the 95% CrI, whereas the 'Rhat' column reports the Gelman-Rubin statistic regarding the convergence. The last column reports the Bayes factor in favour of the null hypothesis (BF01).

2   Analysis without outliers

We also ran our model removing participants who were outliers (results below Quartile 1 − 1.5 * Interquartile range or above Quartile 3 + 1.5 * Interquartile Range) on at least half of the condition for accuracy or for RT (3 ASD - 2 females, 1 male - and 3 TD - 2 females, 1 male, as preregistered on https://osf.io/345jv.

All relevant statistics regarding groups description are set out in *Table 4*.

2.1   Demographical data

2.2   Posterior predictive check

Table 4

Subject demographics – Means (M), standard deviations (SD) and BF01.

	Group: ASD (N = 24)	Group: TD (N = 31)	BF01
Age	33.08 (8.37)	31.68 (7.29)	3.04
Education	14.79 (2.69)	15.90 (2.75)	1.445
FSIQ	126.42 (13.92)	113.84 (15.15)	0.071
VIQ	128.88 (10.63)	120.52 (14.71)	0.394
PIQ	118.08 (13.43)	104.06 (15.14)	0.026
AQ	36.17 (7.11)	15.26 (6.24)	1.026 x 10^-13
Reading	0.10 (0.88)	-0.07 (0.55)	2.61
Stroop Effect	0.45 (0.81)	0.55 (0.77)	3.306

Note. *FSIQ = Full Scale Intelligence Quotient; VIQ = Verbal Intelligence Quotient; PIQ = Performance Intelligence Quotient ; AQ = Autism-Spectrum Quotient.
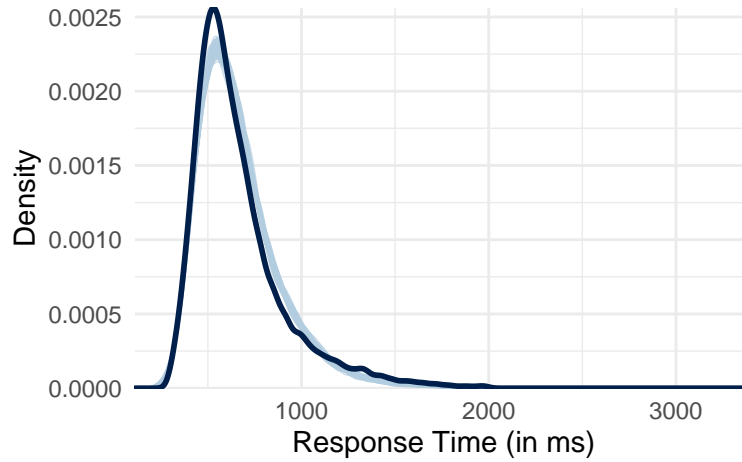

Figure 1. Posterior predictive checking for the BLMM without outliers.

2.3 Results

Table 5

Estimates from the ExGaussian BLMM on RT without outliers.

Predictors	Estimate	SE	Lower	Upper	Rhat	BF01
Intercept	667.48	12.51	643.00	692.00	1.00	NA
Group	52.29	22.12	7.77	94.71	1.00	0.16
Prime	-11.19	2.71	-16.48	-5.87	1.00	0.01
Congruency	-11.43	2.66	-16.72	-6.15	1.00	0.01
Emotion	18.02	5.99	6.27	29.70	1.00	0.11
Group x Prime	1.22	5.11	-8.76	11.18	1.00	9.91
Group x Congruency	0.73	4.74	-8.65	10.06	1.00	10.62
Prime x Congruency	6.09	4.64	-3.08	15.26	1.00	4.70
Group x Emotion	-11.31	8.34	-27.51	5.31	1.00	2.51
Prime x Emotion	1.02	4.47	-7.77	9.85	1.00	11.74
Congruency x Emotion	-0.40	4.55	-9.35	8.46	1.00	11.19
Group x Prime x Congruency	9.28	8.79	-8.12	26.42	1.00	3.33
Group x Prime x Emotion	24.83	8.96	7.31	42.21	1.00	0.15
Group x Congruency x Emotion	11.32	8.84	-5.88	28.78	1.00	2.63
Prime x Congruency x Emotion	-5.08	9.30	-23.30	13.47	1.00	4.80
Group x Prime x Congruency x Emotion	17.19	17.56	-17.54	51.71	1.00	1.86

3   Analysis on Inverse Efficiency Score (IES)

As preregistered, we also fitted models on the IES (= Response Time/Accuracy) to take into account the accuracy.

3.1   Inclusion of different predictors

We compared the 3 following models:

- IBMgcp : IES ~ 1 + GroupC * CongC * PrimeC + (1 | ID)

- IBMgcpe: IES ~ 1 + GroupC * CongC * PrimeC * EmoC+ (1 | ID)

- IBMgcpes : IES ~ 1 + GroupC * CongC * PrimeC * EmoC * SexC + (1 | ID)

- IBMgcpesP: IES ~ 1 + GroupC * CongC * PrimeC * EmoC + SexC + SexC : EmoC + (1 | ID)

Table 6

Expected log pointise predictive density (ELPD) and LOOIC for each of the three BLMM on IES.

model	elpd_diff	se_diff	elpd_loo	se_elpd_loo	p_loo	se_p_loo	looic	se_looic	Model_Weights
IBMgcpe	0.000	0.000	-2,798.330	28.402	87.995	13.260	5,596.661	56.805	0.535
IBMgcpesP	-0.147	0.283	-2,798.478	28.456	88.162	13.335	5,596.955	56.912	0.462
IBMgcpes	-5.391	3.453	-2,803.722	28.166	99.316	13.894	5,607.443	56.332	0.002
IBMgcp	-11.643	6.458	-2,809.973	27.255	79.990	11.494	5,619.946	54.510	0.000

The model `IBMgcpe` seemed to be the best one (see *Table 6*).

3.2   Inclusion of covariates

We then included covariates in the best model. We compared the following models:
- IBMfa : IES ~ 1 + GroupC * CongC * PrimeC * EmoC + FSIQ + AQ + (1 | ID)

- IBMf : IES ~ 1 + GroupC * CongC * PrimeC * EmoC + FSIQ +(1 | ID)

- IBMa : IES ~ 1 + GroupC * CongC * PrimeC * EmoC + AQ +(1 | ID)

- IBMpa : IES ~ 1 + GroupC * CongC * PrimeC * EmoC + PIQ + AQ + (1 | ID)

- IBMp : IES ~ 1 + GroupC * CongC * PrimeC * EmoC + PIQ + (1 | ID)

Table 7
Expected log pointise predictive density (ELPD) and LOOIC for each of the four BLMM on IES, with covariates.

model	elpd_diff	se_diff	elpd_loo	se_elpd_loo	p_loo	se_p_loo	looic	se_looic	Model_Weights
IBMf	0.000	0.000	-2,796.274	27.783	85.749	11.867	5,592.549	55.565	0.222
IBMa	-1.287	0.820	-2,797.561	28.081	87.203	12.582	5,595.122	56.161	0.196
IBMpa	-1.311	1.159	-2,797.586	28.189	87.227	12.787	5,595.171	56.378	0.191
IBMp	-1.399	1.242	-2,797.673	28.225	87.259	12.851	5,595.346	56.451	0.175
IBMfa	-1.742	1.374	-2,798.016	28.362	87.495	12.993	5,596.032	56.724	0.124
IBMgcpe	-2.056	1.509	-2,798.330	28.402	87.995	13.260	5,596.661	56.805	0.091

The model `IBMf`, with FSIQ as a covariate, appear to be the best model (see *Table 7*). We checked the validity of this model with posterior predictive check (cf. *Figure 2*), which appeared to be good.
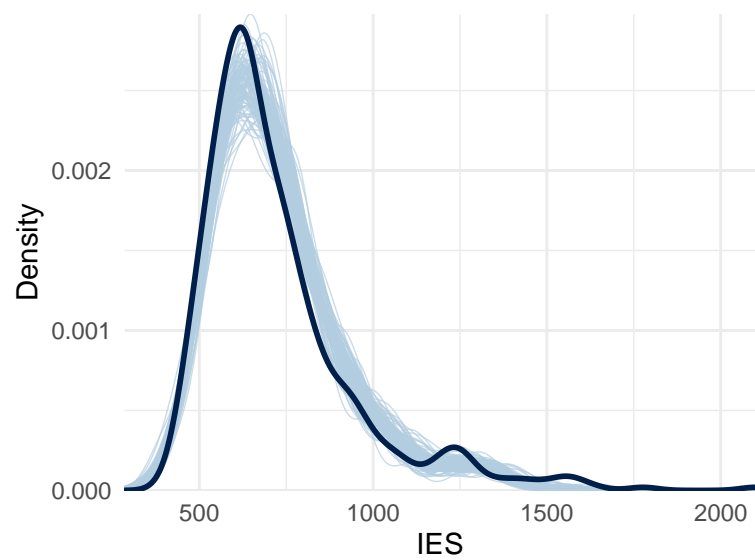
3.3   Posterior predictive check

Figure 2. Posterior predictive checking for BLMM concerning the IES.

3.4 Results

Table 8

Estimates from the ExGaussian BLMM concerning the IES.

| Predictors	Estimate	SE	Lower	Upper	Rhat	BF01
Intercept	730.19	19.53	691.32	767.97	1.00	NA
Group	83.46	31.59	20.32	144.61	1.00	0.061
Congruency	-36.86	5.55	-47.93	-26.20	1.00	5.523 x 10^-17
Prime	-16.09	5.06	-26.09	-6.28	1.00	0.039
Emotion	20.35	5.47	9.75	31.05	1.00	0.015
FSIQ	-41.42	19.11	-78.60	-4.36	1.00	0.255
Group x Congruency	8.64	9.92	-10.97	27.83	1.00	3.475
Group x Prime	-6.75	9.94	-26.23	12.64	1.00	3.931
Prime x Congruency	6.30	9.64	-12.76	25.07	1.00	4.156
Group x Emotion	-28.79	10.71	-50.09	-7.95	1.00	0.123
Congruency x Emotion	1.10	9.70	-17.88	20.03	1.00	5.232
Prime x Emotion	12.70	9.66	-6.34	31.78	1.00	2.202
Group x Prime x Congruency	1.44	18.36	-34.82	37.11	1.00	2.692
Group x Congruency x Emotion	8.27	18.48	-28.12	44.56	1.00	2.446
Group x Prime x Emotion	56.90	18.43	20.70	92.78	1.00	0.025
Prime x Congruency x Emotion	-10.15	18.37	-45.64	25.80	1.00	2.312
Group x Prime x Congruency x Emotion	3.08	31.31	-57.82	64.40	1.00	1.626

4 Visual data exploration
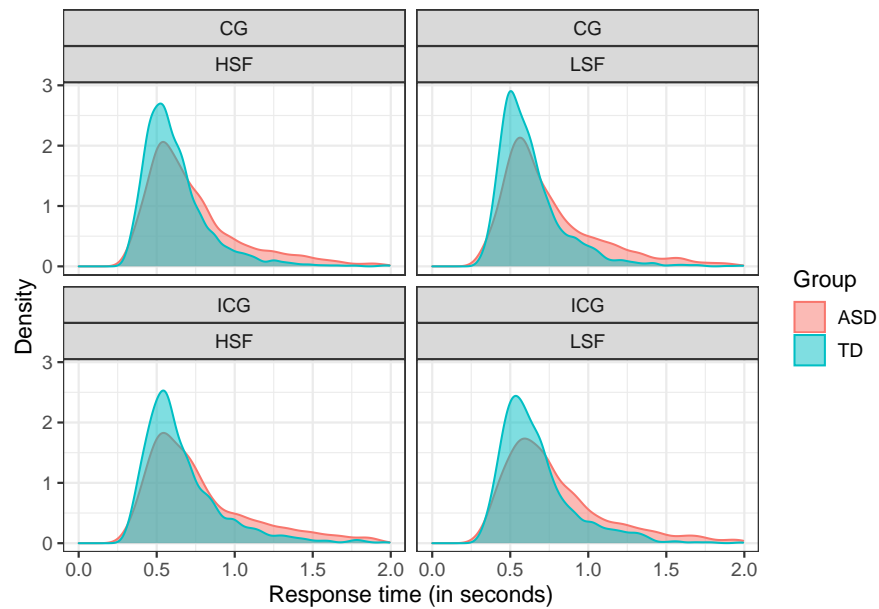
Figure 3. Distribution of RTs per Group, Congruency, and Prime.

Figure 3 shows that the distributions of response times (RTs) are roughly similar across conditions, with slightly slower and more spread RTs in the ASD group as compared

to the TD group. In Figure 4, we plot the distribution of RTs for correct versus incorrect responses according to the characteristics of the participants (Sex and Group). In Figure 5, we make a similar plot, now faceting by condition (i.e., congruency and prime type).
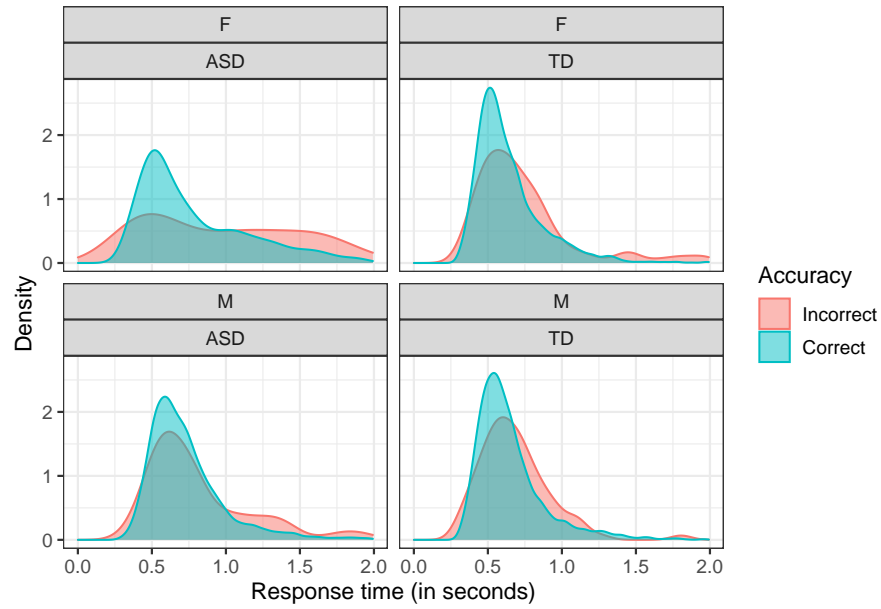

Figure 4. Distribution of RTs for correct and incorrect responses according to Sex and Group. Incorrect responses are in pink and correct responses in blue.
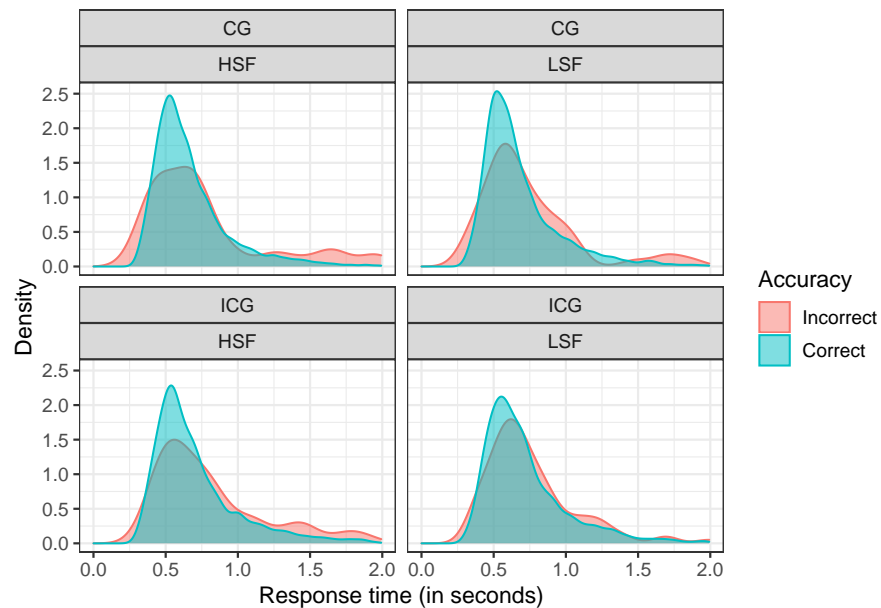

Figure 5. Distribution of RTs for correct and incorrect responses according to Congruency and Prime. Incorrect responses are in pink and correct responses in blue.

5 Diffusion decision modelling

Diffusion decision models (DDM) allow detailed explanations of behaviour in two-alternative forced choice tasks. We know that the data collected in such tasks obey some law-like patterns. For instance, RT distributions are generally positively skewed, with the skewness increasing with task difficulty. We also know that the mean of the RTs is proportional to the standard deviation of the RTs. Increases in the difficulty usually lead to increased RTs and decreased accuracy. Moreover, changes in difficulty also produces regular changes in the distribution of RTs, most notably in its spread but not much in its shape (for a review, see Forstmann, Ratcliff, & Wagenmakers, 2016). Therefore, several models have been proposed to account for the peculiarities of the data coming from such tasks as well as to relate it to the underlying cognitive processes.

5.1 What is a diffusion decision model?

The vanilla (i.e., original) diffusion decision model (the "Wiener model") is a continuous-time evidence accumulation model for binary choice tasks (Ratcliff, 1978). It assumes that in each trial evidence is accumulated in a noisy (diffusion) process by a single accumulator. As shown in Figure 6, evidence accumulation starts at some point (the starting point or "bias") and continues until the accumulator hits one of the two decision bounds in which case the corresponding response is given. The total response time is the sum of the decision time from the accumulation process plus non-decisional components (Vandekerckhove, Verheyen, & Tuerlinckx, 2010; Wabersich & Vandekerckhove, 2014; Wagenmakers, 2009).[1] In other words, this kind of model provides a *decomposition* of RT data that isolates components (of processing) from stimulus encoding to decision so that they can be studied individually (Ratcliff & McKoon, 2008; Wagenmakers, Van Der Maas, & Grasman, 2007).

In sum, the original Wiener model allows decomposing responses to a binary choice tasks and corresponding response times into four latent processes (Singmann, 2017):

- The **drift rate** δ (delta) is the average slope of the accumulation process towards the boundaries (i.e., it represents the average amount of evidence accumulated per unit time). The larger the (absolute value of the) drift rate, the stronger the evidence for the corresponding response option (thus quantifying the "ease of processing").

---

[1] Where diffusion is to be understood as the continuous equivalent of a random-walk.
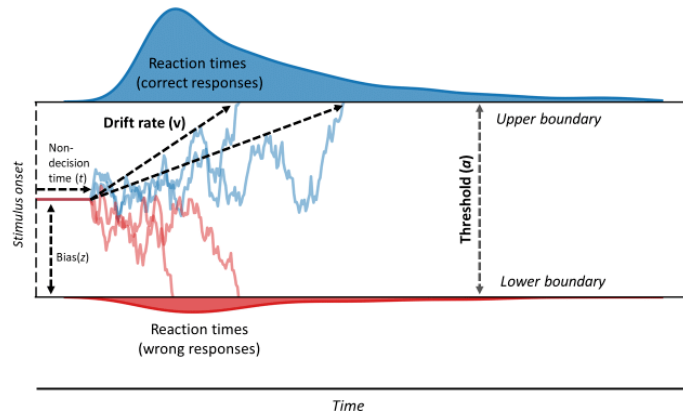
Figure 6. A graphical illustration of the Wiener diffusion model. Figure from Vinding, Lindeløv, Xiao, Chan, & Sørensen (2018).

- The **boundary separation** α (alpha) is the distance between the two decision bounds and can be interpreted as a measure of response caution, with high α meaning high caution.

- The **starting point** (or bias) β (beta) of the accumulation process is a measure of response bias towards one of the two response boundaries (it is sometimes fixed to 0.5 in case of correct vs. incorrect boundaries, but it need not to be).

- The **non-decision time** τ (tau) captures all non-decisional processes such as stimulus encoding and (motor) response processes.

An important decision that has to be made before setting up a model is which parameters are allowed to differ between which conditions. One common constraint of the Wiener model is that the parameters that are set before the evidence accumulation process starts (i.e., boundary separation, starting point, and non-decision time) cannot change based on stimulus characteristics that are not known to the participant before the start of the trial. However, as the response time was recorded from the presentation of the *target*, the participant already had knowledge about the prime when the evidence accumulations began. Thus, the characteristics of the item (i.e., in our case, congruency and spatial characteristics) are only allowed to affect the drift rate, but we compared this model to an augmented model including the nature of the prime (i.e., LSF vs. HSF) and an interaction with Group on the response bias (starting point). Also note that all stimulus-related variables are manipulated within-subject. Thus, the maximal random-effects structure (Barr, Levy, Scheepers, & Tily, 2013) entails corresponding random-effects parameters for each fixed-effect. To set up the

model we need to invoke the `brms::brmsformula()` function and construct one formula for each of the four parameters of the Wiener model.

5.2   Fitting the model

We fitted all models using the `brms` package (Bürkner, 2017).

```r
# defining the model formula (one "linear model" per parameter)
formula <- brmsformula(
  # drift rate (delta)
  RT | dec(ACC) ~ 1 + GroupC * CongC * PrimeC + (1 | ID) + (1 | Target),
  # boundary separation parameter (alpha)
  bs ~ 1 + GroupC + (1 | ID) + (1 | Target),
  # non-decision time (tau)
  ndt ~ 1 + GroupC + (1 | ID) + (1 | Target),
  # starting point or bias (beta)
  bias ~ 1 + GroupC * PrimeC + (1 | ID) + (1 | Target)
  )
# defining the priors
priors <- c(
  # priors for the intercepts
  prior("normal(0, 5)", class = "Intercept"),
  prior("normal(0, 1)", class = "Intercept", dpar = "bs"),
  prior("normal(0, 1)", class = "Intercept", dpar = "ndt"),
  prior("normal(0, 1)", class = "Intercept", dpar = "bias"),
  # priors for the slopes
  prior("normal(0, 1)", class = "b"),
  # priors on the SD of the varying effects
  prior("exponential(1)", class = "sd")
  )
```

We then fit this model below using the `brms::brm()` function. We run eight chains, each for 5000 iterations and using the first 2000 iterations used as warmup (i.e., the first 2000 samples of each chain are discarded from the final analysis). This results in a total of $8 \times (5000 - 2000) = 24000$ samples from the (joint) posterior distribution that will be used for inference.

```r
# specify initial values to help the model start sampling
```

```r
# (with small variation between chains)
chains <- 8 # number of chains
epsilon <- 0.1 # variability in starting value for the NDT intercept
get_init_value <- function(x) list(Intercept_ndt = rnorm(n = 1, mean = x, sd = epsilon) )
inits_drift <- replicate(chains, get_init_value(-3), simplify = FALSE)
# fitting the model
varying_effects_prime <- brm(
  formula,
  data = df,
  # specifying the family and link functions for each parameter
  family = wiener(
    link = "identity", link_bs = "log",
    link_ndt = "log", link_bias = "logit"
    ),
  # comment this line to use default priors
  prior = priors,
  # list of initialisation values
  inits = inits_drift,
  init_r = 0.05,
  warmup = 2000, iter = 5000,
  chains = chains, cores = chains,
  control = list(adapt_delta = 0.99, max_treedepth = 15),
  # saves the model (as .rds) or loads it if it already exists
  file = "models/wiener_varying_effects_prime.rds",
  # needed for hypothesis testing
  sample_prior = TRUE
  )
```

6   DDM: Interpretation of the results

6.1   Model comparison

We first compared the previous model with a similar model including an effect of the prime on the response bias parameter. As response time was recorded from the presentation of the target, it is possible that the prime influenced the response bias toward accurate (or erroneous) responses. Table 9 reports the expected log-pointwise predictive density (ELPD,

a measure of the model's fit), the number of effective parameters (p_WAIC, a measure of the model's complexity) and the WAIC, as well as the difference of these criteria between the two models.

Table 9
Expected log-pointwise predictive density (ELPD) and WAIC for the two DDMs.

model	elpd_diff	se_diff	elpd_waic	se_elpd_waic	p_waic	se_p_waic	waic	se_waic	weight
varying_effects_prime	0.000	0.000	4,123.821	139.639	287.720	11.840	-8,247.641	279.279	0.878
varying_effects	-16.991	6.742	4,106.830	139.704	284.992	11.778	-8,213.659	279.409	0.122

From this table (especially from the last two columns) we see that the varying-effects model including an effect of Prime on the response bias has the lowest WAIC, which can be interpreted by saying that this model is estimated to be the closest to the "true model" in an information-theoretic sense, or equivalently, that it is expected to have the lowest out-of-sample deviance. In the following, we therefore report estimates from this model.

6.2 Interpreting the output

Now that we have fitted the model, we are left with the task of interpreting the output from the best model. As can be seen from the model summary displayed below, the output of the model is a (joint) posterior distribution over all parameters of the model. We can marginalise this joint distribution to obtain the (marginal) posterior distribution on each parameter, which is given by the `summary()` method.

```
# prints a summary of the model
summary(varying_effects_prime)

##  Family: wiener
##   Links: mu = identity; bs = log; ndt = log; bias = logit
## Formula: RT | dec(ACC) ~ 1 + GroupC * CongC * PrimeC + (1 | ID) + (1 | Target)
##          bs ~ 1 + GroupC + (1 | ID) + (1 | Target)
##          ndt ~ 1 + GroupC + (1 | ID) + (1 | Target)
##          bias ~ 1 + GroupC * PrimeC + (1 | ID) + (1 | Target)
##    Data: df (Number of observations: 14420)
## Samples: 8 chains, each with iter = 5000; warmup = 2000; thin = 1;
##          total post-warmup samples = 24000
##
## Group-Level Effects:
## ~ID (Number of levels: 61)
```

```
##                      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)            0.78      0.08     0.64     0.95 1.00     6602    12714
## sd(bs_Intercept)         0.22      0.02     0.17     0.27 1.00     8388    14065
## sd(ndt_Intercept)        0.18      0.02     0.15     0.22 1.00     9810    13731
## sd(bias_Intercept)       0.24      0.04     0.17     0.32 1.00    13276    15996
##
## ~Target (Number of levels: 30)
##                      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## sd(Intercept)            0.17      0.03     0.12     0.24 1.00    11928    17197
## sd(bs_Intercept)         0.01      0.01     0.00     0.02 1.00     9457    13260
## sd(ndt_Intercept)        0.01      0.01     0.00     0.02 1.00     5467     9913
## sd(bias_Intercept)       0.05      0.02     0.01     0.10 1.00     4977     6742
##
## Population-Level Effects:
##                      Estimate Est.Error l-95% CI u-95% CI Rhat Bulk_ESS Tail_ESS
## Intercept                3.01      0.11     2.80     3.23 1.00     4403     8661
## bs_Intercept             0.68      0.03     0.63     0.74 1.00     5326    10354
## ndt_Intercept           -1.24      0.03    -1.29    -1.19 1.00     5319     9449
## bias_Intercept          -0.34      0.04    -0.42    -0.26 1.00    15869    17506
## GroupC                  -0.01      0.20    -0.42     0.38 1.00     4218     7608
## CongC                    0.32      0.03     0.27     0.38 1.00    53999    15341
## PrimeC                  -0.11      0.04    -0.19    -0.03 1.00    29665    20079
## GroupC:CongC            -0.10      0.05    -0.20     0.00 1.00    51192    17442
## GroupC:PrimeC            0.05      0.08    -0.12     0.21 1.00    32073    18817
## CongC:PrimeC             0.06      0.05    -0.04     0.17 1.00    52638    17123
## GroupC:CongC:PrimeC     -0.02      0.11    -0.23     0.19 1.00    48842    16606
## bs_GroupC                0.08      0.06    -0.04     0.20 1.00     4849     9417
## ndt_GroupC               0.18      0.05     0.08     0.28 1.00     4676     9571
## bias_GroupC              0.01      0.08    -0.15     0.17 1.00    15741    17515
## bias_PrimeC              0.16      0.03     0.11     0.21 1.00    29322    20146
## bias_GroupC:PrimeC      -0.02      0.05    -0.13     0.08 1.00    31600    18534
##
## Samples were drawn using sampling(NUTS). For each parameter, Bulk_ESS
## and Tail_ESS are effective sample size measures, and Rhat is the potential
## scale reduction factor on split chains (at convergence, Rhat = 1).
```

We can retrieve the samples from the joint posterior distribution using the

`brms::posterior_samples()` method.

```
# retrieves posterior samples (for all parameters)
varying_effects_prime <- readRDS(file = "models/wiener_varying_effects_prime.rds")
post <- posterior_samples(varying_effects_prime)
# the "post" object has 24.000 rows (number of posterior samples) and
# 396 columns (number of parameters)
dim(post)
```

```
## [1] 24000    396
```

```
# displaying the first six samples for the first 5 parameters
head(post[, 1:5])
```

```
##    b_Intercept b_bs_Intercept b_ndt_Intercept b_bias_Intercept      b_GroupC
## 1     2.904270      0.6586814       -1.275385       -0.3706857  -0.11342708
## 2     2.972417      0.6528011       -1.308810       -0.4090769   0.03222316
## 3     2.868398      0.6492471       -1.292156       -0.2937770  -0.20980861
## 4     3.050407      0.7175195       -1.257030       -0.3241950  -0.02612217
## 5     3.105057      0.7030418       -1.215820       -0.2389826   0.01180252
## 6     3.122955      0.6999282       -1.217508       -0.2114922  -0.03119975
```

Which outputs a matrix with parameters of the model in columns and posterior samples in rows. Let's examine these results for each parameter in more details.

6.2.1 Drift rate. From there, we can study the marginal posterior distribution for each parameter of interest. For instance, Figure 7 represents the posterior distribution of the intercept for the drift rate (i.e., the overall mean value of the drift rate).

```
# retrieves the posterior samples for the average drift rate
average_drift <- post$b_Intercept
# plotting it
plotPost(
  average_drift, showMode = TRUE,
  xlab = expression(paste(beta[0][paste("[", delta, "]")] ) )
  )
```

Figure 7 reveals (consistently with the summary displayed previously) that the most probable value for the intercept of the drift rate (given our model, the priors, and the data) is 3.014, and that there is a 95% probability that the population value for this parameter lies in the interval ranging from 2.799 to 3.234. More interestingly, we can inspect the
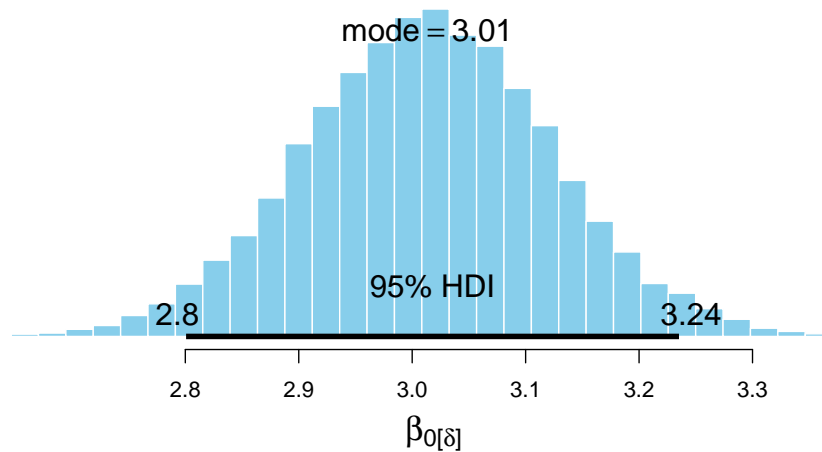
Figure 7. Posterior distribution of the intercept for the drift rate. The mode (i.e., the most probable value) and the 95% credible interval are also displayed.

posterior distribution of the slopes for the effects of `Group`, `Prime`, and `Cong`.

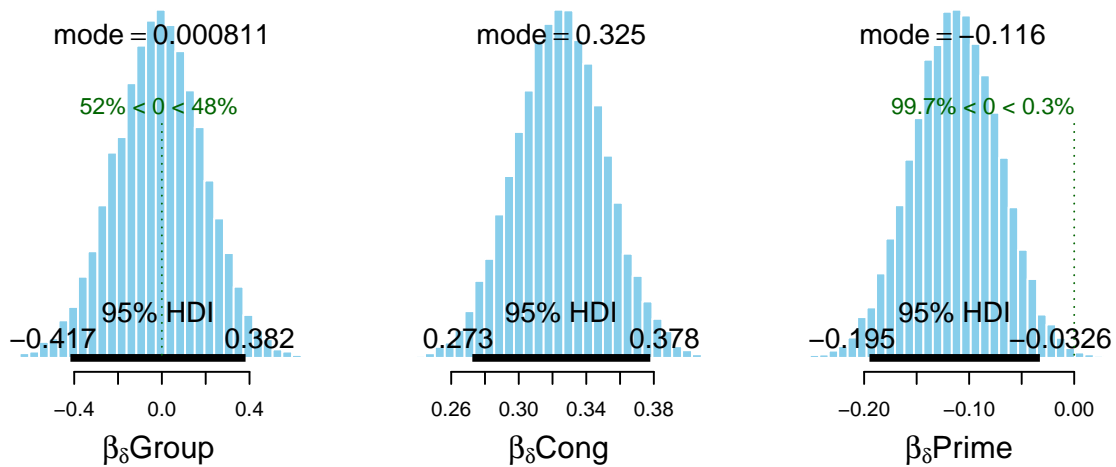*Figure 8.* Posterior distribution of the slopes for the drift rate, for Group, Prime, and Cong (respectively). The mode (i.e., the most probable value) and the 95% credible interval are also displayed.

Figure 8 reveals that `Cong` increased the drift rate whereas `Prime` decreased it (recall that the drift rate can be interpreted as the "ease of processing"),[2] whereas the effect of `Group` is estimated to be quasi null (cf. later hypothesis testing). What about interactions?

Figure 9 reveals that the interaction between `Group` and `Prime` and the double interaction between `Group`, `Cong`, and `Prime` were most probably close to zero. However, the interaction between `Group` and `Cong` seems reliably negative, which can be interpreted as

---

[2] As `Cong` and `Prime` were coded using sum contrasts (-0.5 vs. 0.5), these effects can be interpreted as follows: going from incongruent to congruent stimuli or to HSF to LSF stimuli is associated with "more" ease of processing.
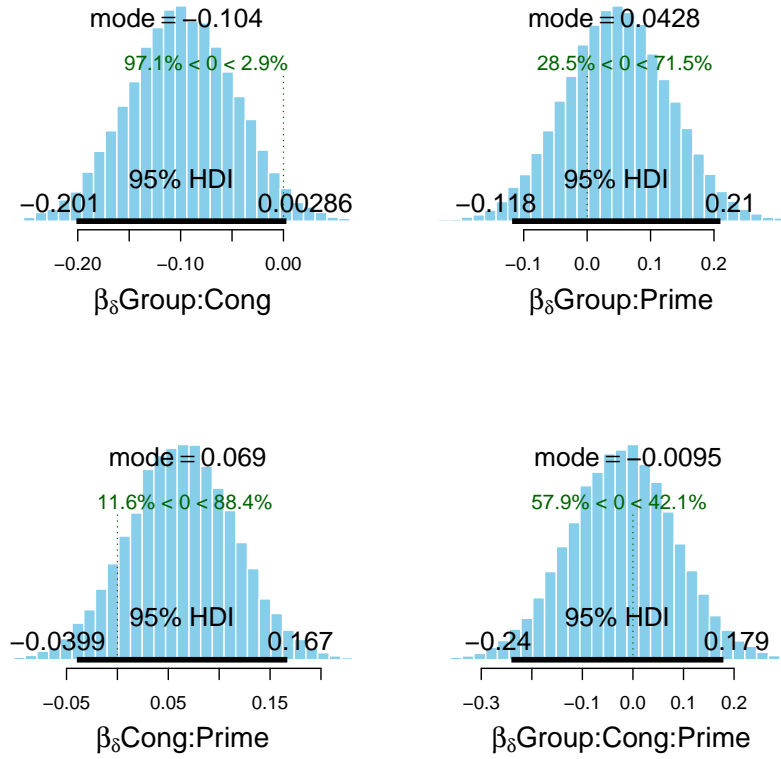
Figure 9. Posterior distribution of the interactions slopes for the drift rate. The mode (i.e., the most probable value) and the 95% credible interval are also displayed.

follows: going from TD to ASD is associated with a decrease in the effect of `Cong` (which was positive) on the drift rate. In other words, whereas the main effect of `Cong` indicates that congruent stimuli are easier to process than incongruent stimuli, this interaction indicates that this effect is less pronounced in the ASD group than in the TD group. The interaction between `Cong` and `Prime` can be interpreted similarly, where going from incongruent to congruent stimuli increase the effect of `Prime` (which was positive) on the drift rate (and reciprocally).

6.2.2 Boundary separation. Recall that the boundary separation parameter can be interpreted as a measure of response caution (with high α corresponding to high response caution), and that the linear model for this parameter is on the log scale (i.e., we used a log link function): $\log(\alpha_i) = \beta_0 + \beta_1 \cdot \text{Group}_i$.

Therefore, we have to apply the inverse link function (i.e., $\exp(\cdot)$) to the parameter to be able to interpret it. Taking $\exp(\beta_1)$ gives the proportional change in the value of the boundary-separation parameter when we go from the TD group to the ASD group (see upper right panel of Figure 10). In our case, $\exp(\beta_1) \approx 1.083$, which means that going from TD to ASD leads to an increase of approximately 7% in the value of the boundary-
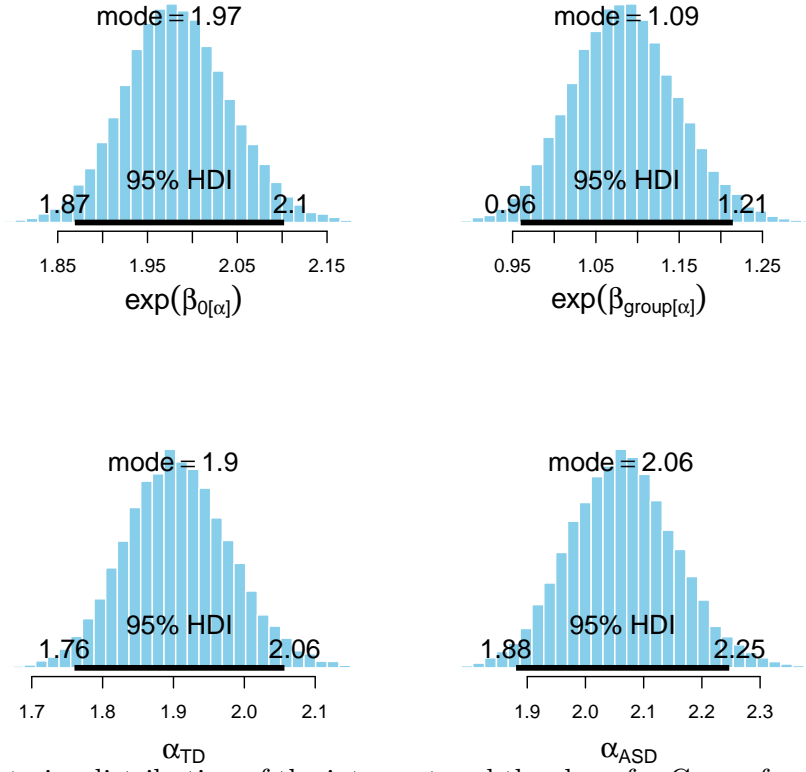
Figure 10. Posterior distribution of the intercept and the slope for Group for the boundary separation. The upper two distributions are the distributions of the intercept and the slope (on the linear scale). The lower two plots are the posterior distributions of the boundary separation parameter in the TD and ASD groups, respectively (on the linear scale).

separation parameter. In other words, response caution seems higher in the ASD group (see lower right panel of Figure 10) than in the TD group (see lower left panel of Figure 10).

6.2.3 Starting point. The starting point is a measure of response bias towards one of the two response boundaries (here, correct vs. incorrect) and is bounded between 0 and 1. The linear model for this parameter is on the logit (log-odds) scale: $\log(\frac{\beta_i}{1-\beta_i}) = \beta_0 + \beta_1 \cdot \text{Group}_i$. Therefore, we have to apply the inverse link function (i.e., $\text{logit}^{-1}(\beta_i) = \text{logistic}(\beta_i) = \frac{1}{1+\exp(-\beta_i)} = \frac{\exp(\beta_i)}{\exp(\beta_i)+1}$) to the parameter to be able to interpret it on its natural scale (i.e., between 0 and 1).

Taking the exponential of the slope gives an odds ratio of approximately 0.99, which indicates that the odds of producing a correct answer (vs. an incorrect one) are approximately the same in both groups. This can also be read from the posterior distribution of the bias parameter in the ASD group, when compared with the posterior distribution of the bias parameter in the TD group. We can also plot the posterior distribution of the difference between ASD and TD (cf. Figure 12).
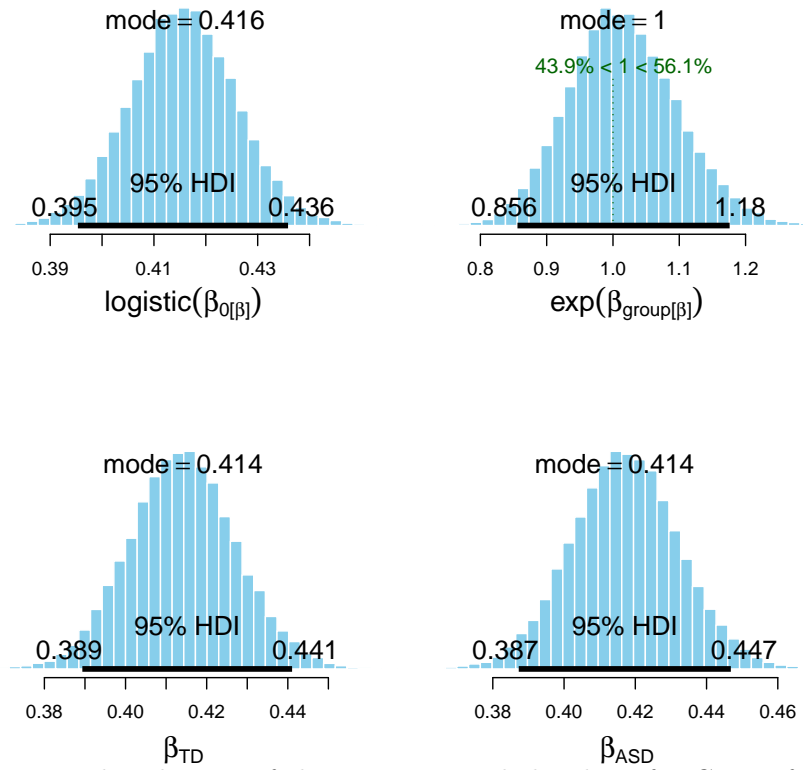
Figure 11. Posterior distribution of the intercept and the slope for Group for the sarting point (bias). The upper two distributions are the distributions of the intercept (on the natural scale of the parameter) and the exponential of the slope (giving an odds ratio). The lower two plots are the posterior distributions of the estimated statting points in the TD and ASD groups, respectively (on the linear scale).
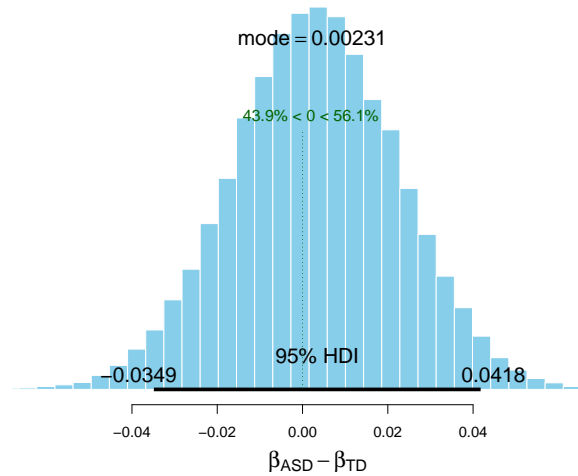
Figure 12. Posterior distribution of the difference between the estimated starting point in the ASD group and the estimated starting point in the TD group.

As shown in Figure 13, the effect of `Prime` was positive, which means that HSF

(vs. LSF) primes were associated with a "bias" towards accurate responses. No interaction effect was found between `Group` and `Prime`.
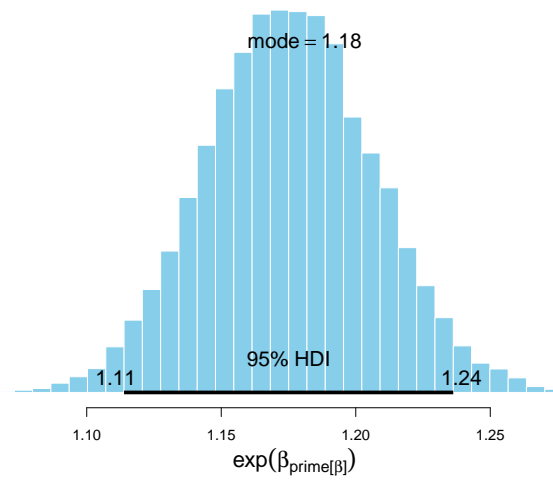
Figure 13. Posterior distribution of the odds ratio between LSF and HSF primes on the starting point.

6.2.4 Non-decision time. Recall that the non-decision time parameter can be interpreted as a measure of the time used by non-decisional processes such as stimulus encoding or motor response, and that the linear model for this parameter is on the log scale (i.e., we used a log link function): $\log(\tau_i) = \beta_0 + \beta_1 \cdot \text{Group}_i$.

Figure 14. Posterior distribution of the intercept and the slope for Group for the non-decision time. The upper two distributions are the distributions of the intercept and the slope (on the linear scale). The lower two plots are the posterior distributions of the non-decision time in the TD and ASD groups, respectively (on the linear scale).
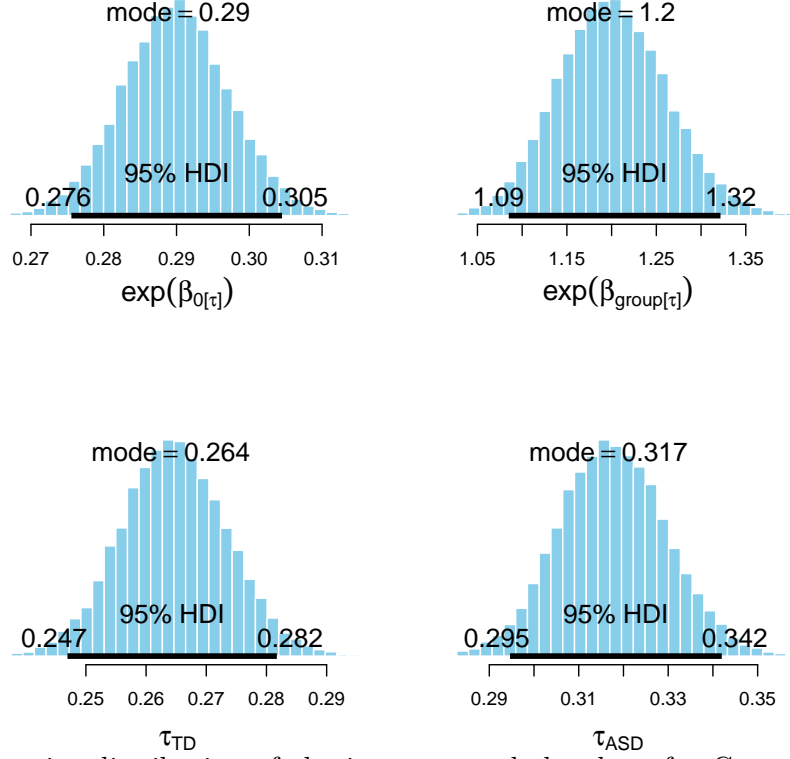
Therefore, we have to apply the inverse link function (i.e., $\exp(\cdot)$) to the parameter to be able to interpret it. Taking $\exp(\beta_1)$ gives the proportional change in the value of the non-decision time parameter when we go from the TD group to the ASD group. In our case, $\exp(\beta_1) \approx 1.24$ which means that going from TD to ASD leads to an increase of approximately 24% of the non-decision time parameter value. In other words, non-decisional processes seem to take longer in the ASD group than in the TD group.

6.3   Evaluating the model

One way of evaluating the model is to evaluate its predictions. If the model is a good description of the (true) process that generated the observed data, then it should be able to generate data that looks like the observed data. The process of generating data from the estimated posterior distribution is called *posterior predictive checking* and can be used in many different ways using the `pp_check()` method (Gabry, Simpson, Vehtari, Betancourt, & Gelman, 2019). In Figure 15, we depict the distribution of the raw data along with the distribution of ten simulated datasets.

```
pp_check(varying_effects_prime, nsamples = 10) +
  theme_bw(base_size = 12) +
  labs(x = "Response time (in seconds)", y = "Density")
```
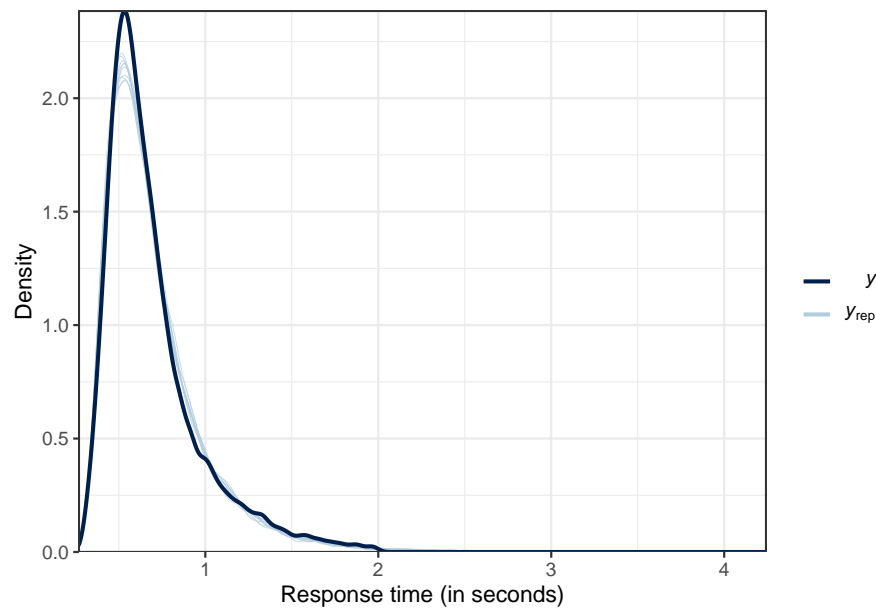

Figure 15. Posterior predictive checking. The dark blue line represents the distribution of raw data whereas light blue lines represent data simulated from the posterior distribution.

As can be seen from Figure 15, the model seems pretty good at simulating data that looks like the observed data. From this predictive/sampling distribution (i.e., the distribution of simulated data sets), so-called "Bayesian p-values" can be computed to quantify the compatibility between the observed data and the proposed model.

6.4   Hypothesis testing

We can test any arbitrary hypothesis using the `brms::hypothesis()` method, which is computing a Bayes factor via the Savage-Dickey method (Wagenmakers, Lodewyckx, Kuriyal, & Grasman, 2010), which consists in comparing the posterior probability density to the prior probability density for some hypothesised value for the parameter of interest (e.g., $\theta = 0$).

```r
# Plotting the Savage-Dickey Bayes factor
# testing whether the interaction Group * Congruency is equal to 0
hyp <- hypothesis(varying_effects_prime, "GroupC:CongC = 0")
# prints the output
print(hyp)

## Hypothesis Tests for class b:
##           Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio Post.Prob
## 1 (GroupC:CongC) = 0     -0.1      0.05     -0.2        0       3.16      0.76
##   Star
## 1
## ---
## 'CI': 90%-CI for one-sided and 95%-CI for two-sided hypotheses.
## '*': For one-sided hypotheses, the posterior probability exceeds 95%;
## for two-sided hypotheses, the value tested against lies outside the 95%-CI.
## Posterior probabilities of point hypotheses assume equal prior probabilities.

# plotting it
data.frame(posterior = hyp$samples$H1, prior = hyp$prior_samples$H1) %>%
  gather(type, value) %>%
  ggplot(aes(x = value, fill = type) ) +
  geom_vline(xintercept = 0, linetype = 3, alpha = 1) +
  geom_area(stat = "density", alpha = 0.8, position = "identity") +
  theme_bw(base_size = 12) +
  labs(x = expression(beta[group:congruency]), y = "Probability density") +
  scale_fill_brewer(palette = "Dark2", direction = -1) +
  theme(legend.title = element_blank() ) +
  coord_cartesian(xlim = c(-2, 2) )
```

The resulting Bayes factor (called "Evid. Ratio" in the output) may be interpreted as follows: the observed data is 4.09 more likely under the hypothesis of null effect than
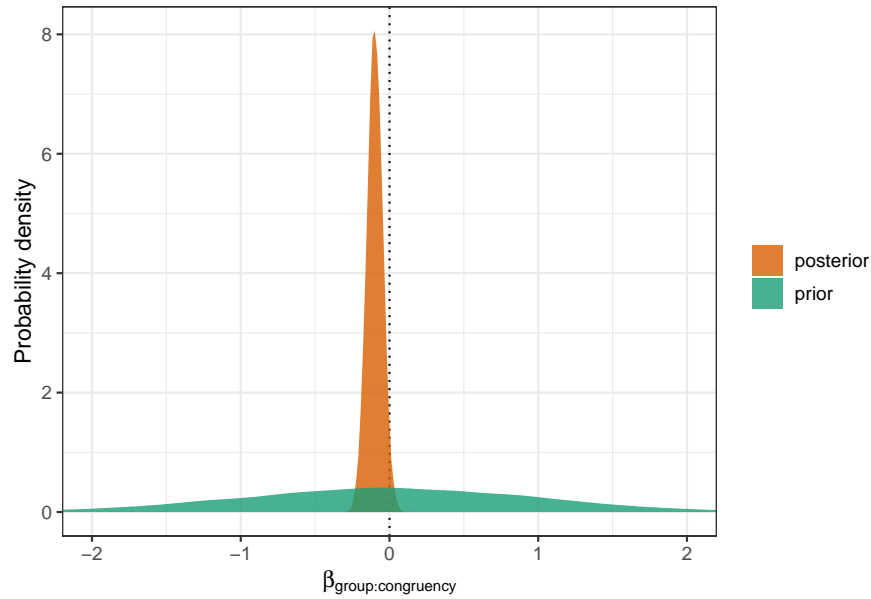
Figure 16. Hypothesis testing via the Savage-Dickey method. The resulting Bayes factor (BF) is the ratio of the height (i.e., the density probability) of the posterior versus prior distribution at some value of interest for the parameter (here it is 0).

under the hypothesis of a non-null effect. Alternatively, the BF can be interpreted as an *updating factor*, indicating by "how much" we should update our *prior odds* (the ratio of the a priori probability of H_0 versus H_1) to convert them into *posterior odds* (the ratio of the a posteriori probability of H_0 versus H_1). Note however that the BF is highly dependent on the prior put on the parameter. In this example, we see that despite the fact that the posterior distribution is "less centered" on 0 than the prior distribution, because of the vagueness of the prior distribution, the posterior distribution still allocates more probability density to 0 than the prior distribution does.

Finally, Table 10 reports the estimates and associated credible intervals and BF01s for all slopes in the hierarchical DDM. To sum up our results, these estimates suggest that `Cong` increasew the drift rate, while `Prime` seems to decrease it, which means that congruent (vs. incongruent) and LSF (vs. HSF) stimuli increase the ease of processing of the stimulus. The positive effect of `Group` on the non-decision time parameter suggests non-decision related processes (such as the stimulus processing or motor processes) are different according to the group (i.e., they take longer in the ASD group than in the TD group). Finally, the effect of `Prime` on the starting point suggests that HSF (vs. LSF) primes were associated with an a priori "bias" towards accurate responses.

Table 10

Estimates and BFs for the slopes from the hierarchical DDM.

Term	Estimate	SE	Lower	Upper	Rhat	BF01
GroupC	-0.012	0.204	-0.416	0.383	1.002	4.872
CongC	0.325	0.027	0.273	0.378	1.000	5.533*10^-17
PrimeC	-0.113	0.042	-0.195	-0.032	1.000	0.516
GroupC:CongC	-0.099	0.053	-0.201	0.003	1.000	3.162
GroupC:PrimeC	0.048	0.084	-0.116	0.213	1.000	9.958
CongC:PrimeC	0.063	0.053	-0.040	0.167	1.000	8.975
GroupC:CongC:PrimeC	-0.022	0.106	-0.233	0.187	1.001	9.153
bs_GroupC	0.079	0.060	-0.039	0.196	1.001	6.715
ndt_GroupC	0.182	0.051	0.084	0.282	1.001	0.011
bias_GroupC	0.012	0.081	-0.147	0.169	1.001	12.058
bias_PrimeC	0.161	0.027	0.109	0.213	1.000	1.618*10^-17
bias_GroupC:PrimeC	-0.022	0.054	-0.127	0.084	1.000	17.235

Note. For each slope (for each line), the first two columns represent the estimated most probable value and its standard error (SE). The 'Lower' and 'Upper' columns contain the lower and upper bounds of the 95% CrI, whereas the 'Rhat' column reports the Gelman-Rubin statistic. The last column reports the Bayes factor in favour of the null hypothesis (BF01).

7 Acknowledgments

8   Session information

```
sessionInfo()
```

```
## R version 4.0.3 (2020-10-10)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Catalina 10.15.4
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRblas.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.0/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## attached base packages:
## [1] parallel  stats     graphics  grDevices utils     datasets  methods
## [8] base
##
## other attached packages:
##  [1] emmeans_1.5.3         BayesFactor_0.9.12-4.2 Matrix_1.2-18
##  [4] coda_0.19-4           flextable_0.6.1        ggpubr_0.4.0
##  [7] kableExtra_1.3.1      qwraps2_0.5.0          car_3.0-10
## [10] carData_3.0-4         reshape2_1.4.4         lattice_0.20-41
## [13] plyr_1.8.6            here_1.0.1             ggrepel_0.8.2
## [16] patchwork_1.1.0       BEST_0.5.2             HDInterval_0.2.2
## [19] brms_2.14.4           Rcpp_1.0.6             glue_1.4.2
## [22] knitr_1.31            papaja_0.1.0.9997      tinylabels_0.1.0
## [25] forcats_0.5.1         stringr_1.4.0          dplyr_1.0.3
## [28] purrr_0.3.4           readr_1.4.0            tidyr_1.1.2
## [31] tibble_3.0.6          ggplot2_3.3.3          tidyverse_1.3.0
##
## loaded via a namespace (and not attached):
##  [1] uuid_0.1-4            readxl_1.3.1           backports_1.2.1
##  [4] systemfonts_0.3.2     igraph_1.2.6           splines_4.0.3
##  [7] crosstalk_1.1.1       TH.data_1.0-10         rstantools_2.1.1
## [10] inline_0.3.17         digest_0.6.27          htmltools_0.5.1.1
```

```
##  [13] rsconnect_0.8.16    magrittr_2.0.1      openxlsx_4.2.3
##  [16] modelr_0.1.8        RcppParallel_5.0.2  matrixStats_0.58.0
##  [19] officer_0.3.15      xts_0.12.1          sandwich_3.0-0
##  [22] prettyunits_1.1.1   colorspace_2.0-0    rvest_0.3.6
##  [25] haven_2.3.1         xfun_0.20           callr_3.5.1
##  [28] crayon_1.3.4        jsonlite_1.7.2      lme4_1.1-26
##  [31] survival_3.2-7      zoo_1.8-8           gtable_0.3.0
##  [34] RWiener_1.3-3       MatrixModels_0.4-1  webshot_0.5.2
##  [37] V8_3.4.0            pkgbuild_1.2.0      rstan_2.21.2
##  [40] abind_1.4-5         scales_1.1.1        mvtnorm_1.1-1
##  [43] DBI_1.1.1           rstatix_0.6.0       miniUI_0.1.1.1
##  [46] viridisLite_0.3.0   xtable_1.8-4        foreign_0.8-80
##  [49] stats4_4.0.3        StanHeaders_2.21.0-7 DT_0.17
##  [52] htmlwidgets_1.5.3   httr_1.4.2          threejs_0.3.3
##  [55] RColorBrewer_1.1-2  ellipsis_0.3.1      farver_2.0.3
##  [58] pkgconfig_2.0.3     loo_2.4.1           dbplyr_2.0.0
##  [61] labeling_0.4.2      tidyselect_1.1.0    rlang_0.4.10
##  [64] later_1.1.0.1       munsell_0.5.0       cellranger_1.1.0
##  [67] tools_4.0.3         cli_2.3.0           generics_0.1.0
##  [70] broom_0.7.4.9000    ggridges_0.5.3      evaluate_0.14
##  [73] fastmap_1.1.0       yaml_2.2.1          processx_3.4.5
##  [76] fs_1.5.0            zip_2.1.1           pbapply_1.4-3
##  [79] nlme_3.1-151        mime_0.9            projpred_2.0.2
##  [82] xml2_1.3.2          compiler_4.0.3      bayesplot_1.8.0
##  [85] shinythemes_1.2.0   rstudioapi_0.13     gamm4_0.2-6
##  [88] curl_4.3            ggsignif_0.6.0      reprex_1.0.0
##  [91] statmod_1.4.35      stringi_1.5.3       ps_1.5.0
##  [94] Brobdingnag_1.2-6   gdtools_0.2.2       nloptr_1.2.2.2
##  [97] markdown_1.1        shinyjs_2.0.0       vctrs_0.3.6
## [100] pillar_1.4.7        lifecycle_0.2.0     bridgesampling_1.0-0
## [103] estimability_1.3    data.table_1.13.4   httpuv_1.5.5
## [106] R6_2.5.0            bookdown_0.21       promises_1.1.1
## [109] rio_0.5.16          gridExtra_2.3       rjags_4-10
## [112] codetools_0.2-18    boot_1.3-25         colourpicker_1.1.0
## [115] MASS_7.3-53         gtools_3.8.2        assertthat_0.2.1
## [118] rprojroot_2.0.2     withr_2.4.1         shinystan_2.5.0
```

```
## [121] multcomp_1.4-15    mgcv_1.8-33       hms_1.0.0
## [124] grid_4.0.3         minqa_1.2.4       rmarkdown_2.6
## [127] shiny_1.6.0        lubridate_1.7.9.2 base64enc_0.1-3
## [130] dygraphs_1.1.1.6
```

References

Aust, F., & Barth, M. (2020). *papaja: Create APA manuscripts with R Markdown.* Retrieved from https://github.com/crsh/papaja

Barr, D. J., Levy, R., Scheepers, C., & Tily, H. J. (2013). Random effects structure for confirmatory hypothesis testing: Keep it maximal. *Journal of Memory and Language*, *68*(3), 255–278. https://doi.org/10.1016/j.jml.2012.11.001

Bürkner, P.-C. (2017). brms: An R package for Bayesian multilevel models using Stan. *Journal of Statistical Software*, *80*(1), 1–28. https://doi.org/10.18637/jss.v080.i01

Forstmann, B. U., Ratcliff, R., & Wagenmakers, E.-J. (2016). Sequential Sampling Models in Cognitive Neuroscience: Advantages, Applications, and Extensions. *Annual Review of Psychology*, *67*, 641–666. https://doi.org/10.1146/annurev-psych-122414-033645

Gabry, J., Simpson, D., Vehtari, A., Betancourt, M., & Gelman, A. (2019). Visualization in Bayesian workflow. *Journal of the Royal Statistical Society: Series A (Statistics in Society)*, *182*(2), 389–402. https://doi.org/10.1111/rssa.12378

R Core Team. (2019). *R: A language and environment for statistical computing.* Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*(2), 59–108. https://doi.org/10.1037/0033-295X.85.2.59

Ratcliff, R., & McKoon, G. (2008). The diffusion decision model: Theory and data for two-choice decision tasks. *Neural Computation*, *20*(4), 873–922. https://doi.org/10.1162/neco.2008.12-06-420

Singmann, H. (2017). Diffusion/Wiener Model Analysis with brms – Part I: Introduction and Estimation. Retrieved from http://singmann.org/wiener-model-analysis-with-brms-part-i/

Vandekerckhove, J., Verheyen, S., & Tuerlinckx, F. (2010). A crossed random effects diffusion model for speeded semantic categorization decisions. *Acta Psychologica*, *133*(3), 269–282. https://doi.org/10.1016/j.actpsy.2009.10.009

Vinding, M., Lindeløv, J. K., Xiao, Y., Chan, R. C. K., & Sørensen, T. A. (2018). *Volition in Prospective Memory: Evidence against differences in recalling free and fixed delayed intentions.* https://doi.org/10.31234/osf.io/hsrbt

Wabersich, D., & Vandekerckhove, J. (2014). The RWiener Package: An R Package Providing Distribution Functions for the Wiener Diffusion Model. *The R Journal*, *6*(1), 49–56. Retrieved from https://journal.r-project.org/archive/2014/RJ-2014-005/index.html

Wagenmakers, E.-J. (2009). Methodological and empirical developments for the Ratcliff diffusion model of response times and accuracy. *European Journal of Cognitive Psychology*, *21*(5), 641–671. https://doi.org/10.1080/09541440802205067

Wagenmakers, E.-J., Lodewyckx, T., Kuriyal, H., & Grasman, R. (2010). Bayesian hypothesis testing for psychologists: A tutorial on the Savage–Dickey method. *Cognitive Psychology*, *60*(3), 158–189. https://doi.org/10.1016/j.cogpsych.2009.12.001

Wagenmakers, E.-J., Van Der Maas, H. L. J., & Grasman, R. P. P. P. (2007). An EZ-diffusion model for response time and accuracy. *Psychonomic Bulletin & Review*, *14*(1), 3–22. https://doi.org/10.3758/BF03194023

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., … Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Xie, Y. (2015). *Dynamic documents with R and knitr* (2nd ed.). Boca Raton, Florida: Chapman; Hall/CRC. Retrieved from https://yihui.org/knitr/