Supplementary materials to 'Interaction between orthographic and graphomotor constraints in learning to write'

Jérémy Danna[1], Marieke Longcamp[1], Ladislas Nalborczyk[1, 2], Jean-Luc Velay[1], Claire Commengé[2], & Marianne Jover[3]

[1] Aix Marseille Univ, CNRS, LNC, Marseille, France
[2] Aix Marseille Univ, CNRS, LPC, Marseille, France
[3] Aix Marseille Univ, PSYCLE, Aix-en-Provence, France

Contents

1 Visual data exploration

1.1 Univariate raw data exploration

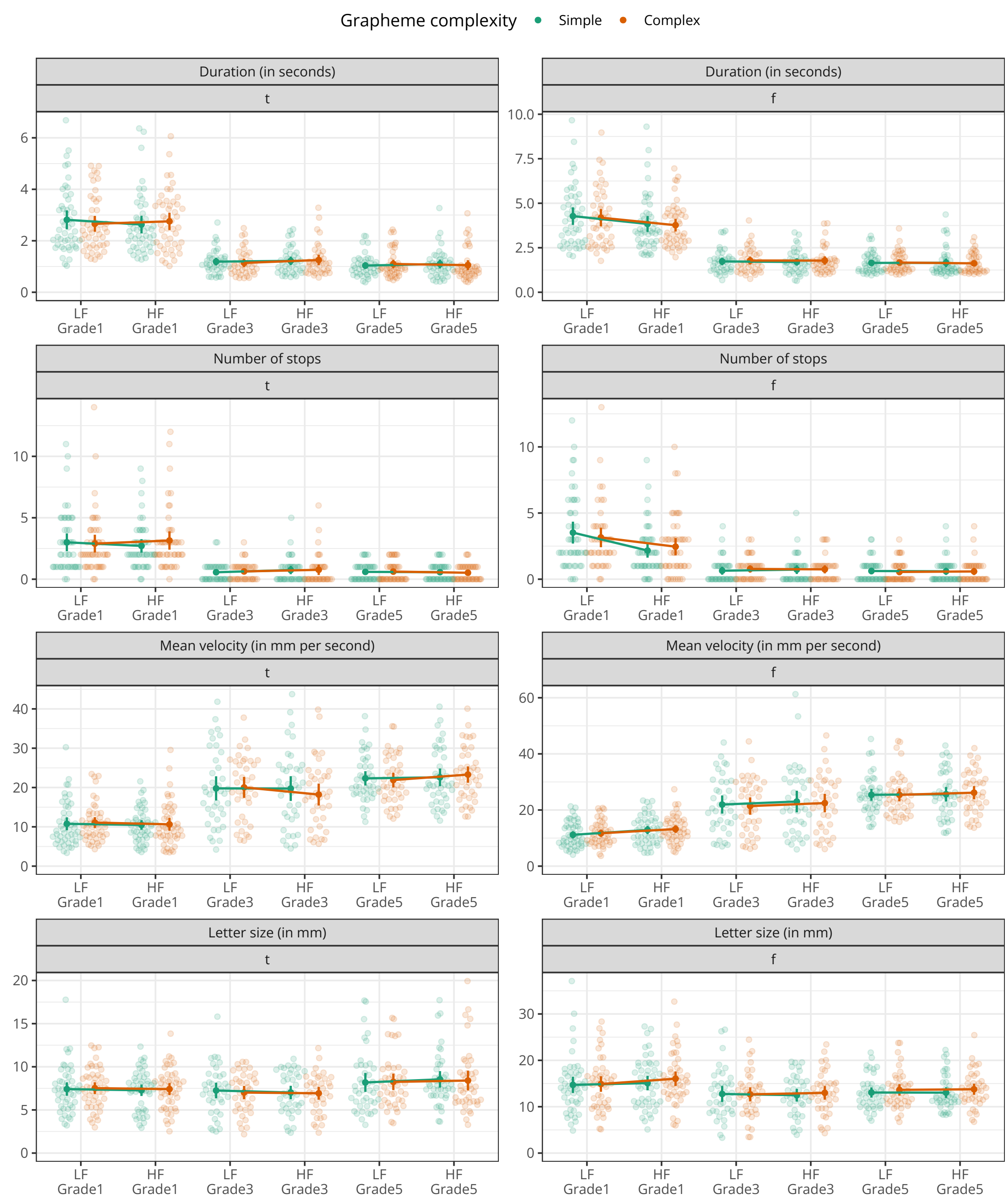

Figure 1. Effect of grade, word frequency, and grapheme complexity on the total duration, the number of stops, the mean velocity, and the letter size. The error bars represent the 95% confidence intervals of the mean (assuming a Gaussian distribution).

Figure 1 shows the effect of grade, word frequency, and grapheme complexity on the letter duration, the number of stops, the mean velocity, and the letter size. This figure suggests that the average duration (in seconds) seems to decrease monotonically with grade. The number of stops also seems to decrease with grade, with most trials for children from grade 2 being associated with no stop.

1.2 Bivariate correlations by grade

Figure 2 shows the overall and by-grade Spearman correlation between each pair of variables. This figure reveals medium to strong positive and negative correlations between each pair of variable. These relations are sometimes non-linear (e.g., between duration and mean velocity), hence the use of Spearman (rather than Pearson) correlation coefficients.

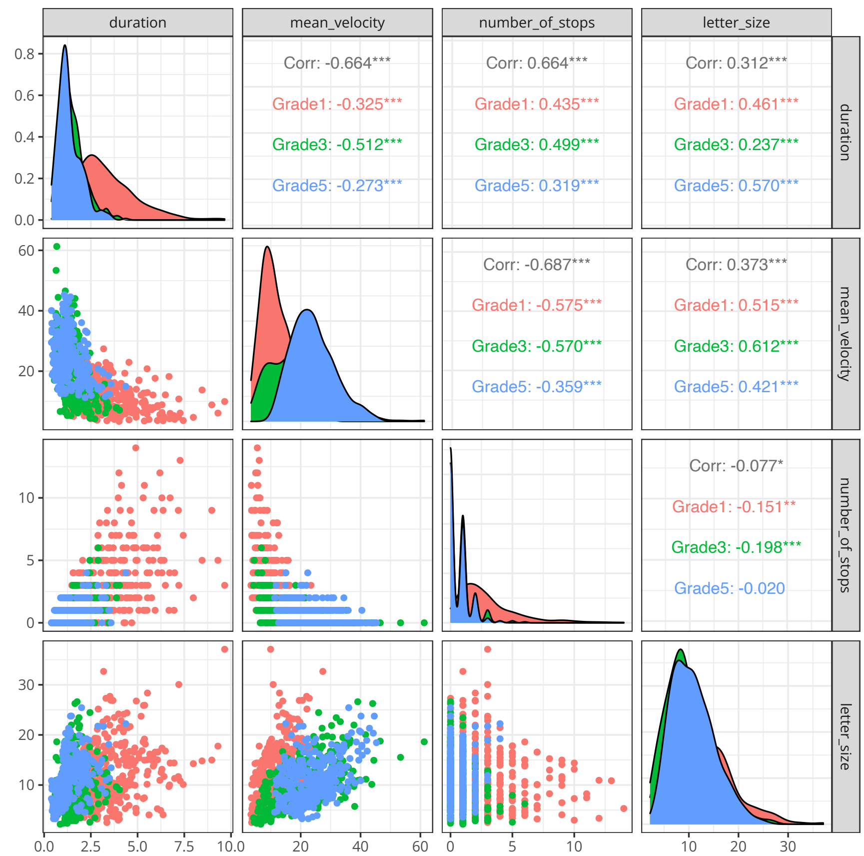

Figure 2. Overall and by-grade Spearman correlation between each pair of measured variables.

2 Bayesian multilevel modelling

2.1 Modelling positive-only values

A dominant feature of durations (or response times) is that their distribution is generally positively skewed, with the spread and/or the skewness increasing with task difficulty (for review, see for instance 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. We discuss below why using Gaussian models for this kind of data is generally not a sensible idea and describe our approach in more details. We follow a general "Bayesian workflow" by building our model in an iterative manner and by motivating and validating each modelling choice (for more details, see for instance Gelman et al., 2020).

We first fitted a Bayesian multilevel (also known as "mixed-effects") Gaussian multivariate (i.e., with multiple outcomes) model. One way of evaluating this model is to evaluate its predictions. If this model is a good description of the 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 3, we depict the distribution of the raw data along with the distribution of 100 simulated datasets.

This figure reveals that the Gaussian model fails to account for the peculiarities of the data at hand. For instance, it systematically fails to predict the right-skew of all four variables, and more dramatically, sometimes predicts negatives values for these variables, although they are strictly positive. Moreover, using a Gaussian distribution to model the number of stops also leads to nonsensical predictions as the number of stops is necessarily a positive integer (whereas the Gaussian distribution can produce any real number), as illustrated in the upper right panel of Figure 3.

2.2 Shifted-lognormal regression model

A useful description of RTs or durations should be able to account for the effects of the difficulty of the task, as well as changes in shift and spread of the distribution. The Log-normal, Ex-Gaussian, or Weibull distributions often provide a good fit to these data, but their parameters are difficult to interpret in terms of difficulty, shift, or spread (i.e., these distributions do not have straightforward interpretable parameters). In contrast, the shifted log-normal distribution has parameters that can easily be interpreted in terms of

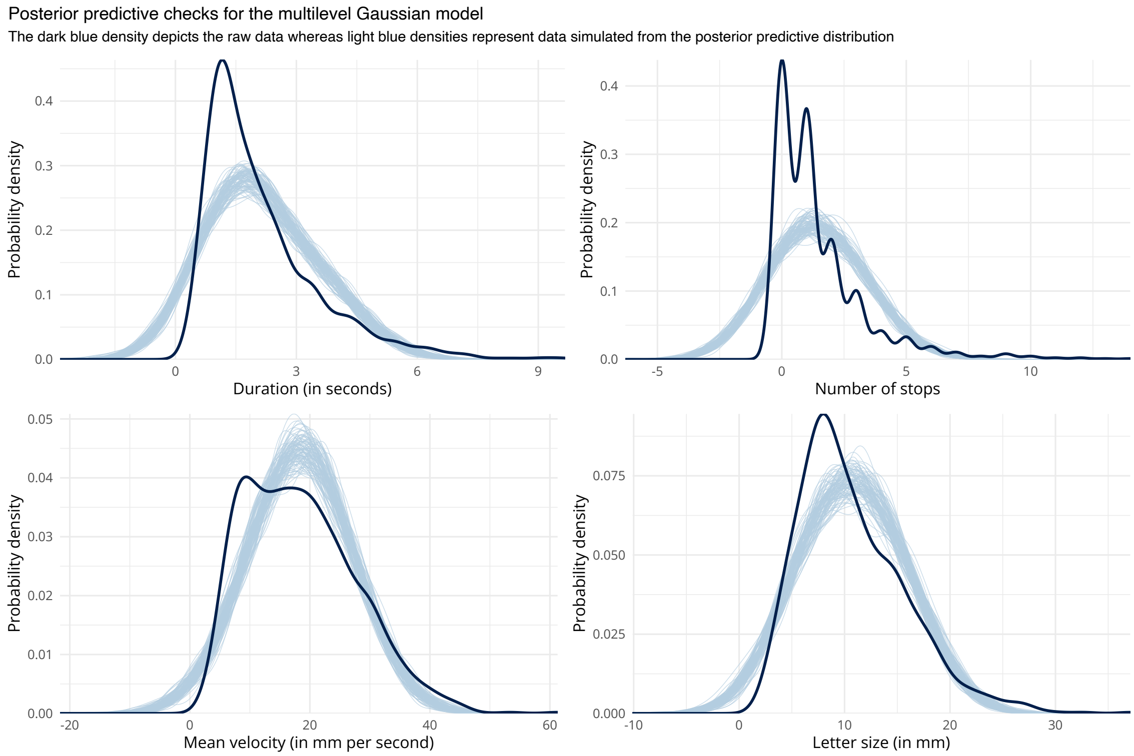

Figure 3. Posterior predictive checks for the multilevel Gaussian model. The dark blue density depicts the raw data whereas light blue densities represent data simulated from the posterior predictive distribution.

difficulty, shift, and spread.

The log-normal distribution is called "log-normal" because the parameters are the mean and standard deviation of the log-transformed response, which is assumed to be a normal (Gaussian) distribution. The shifted log-normal distribution is then described by three parameters:

- μ (mu, difficulty): the mean of the log-normal distribution. The median duration is given by shift $+ \exp(\mu)$.
- σ (sigma, scale): the standard deviation of the log-normal distribution. Increases the mean but not the median of μ.
- shift (ndt) indicates the time of the earliest possible response. When shift $= 0$, the shifted log-normal distribution correspond to the conventional log-normal distribution with two parameters.

Regression models using this family usually aim to predict μ, the mean of the log-normal distribution. We employ this strategy as well, while noticing that both σ and shift could be allowed to vary across conditions as well.

2.3 Poisson regression model

Whereas the previous models are able to take into account the right-skew of the four variables we are interested in, they are still not able to make proper predictions with regards to the number of stops (because the log-normal distribution is continuous). Yet a better model could be fitted by picking up a discrete probability distribution defined on the positive integer real. The Poisson regression model is appropriate for modelling discrete counts of events (e.g., the number of stops) that happen in a fixed interval of space or time with no upper bound. The Poisson model is simpler than the Gaussian or the lognormal one because it has only one parameter λ that describes its shape. The parameter λ is the expected value of the outcome y (and also its expected variance). However, we need a link function to relate the predictors with the parameter λ and to ensure that λ is always positive. We use the conventional logarithmic link function, resulting in the following linear model:

$$
\begin{aligned}
y_i &\sim \text{Poisson}(\lambda_i) \\
\log(\lambda_i) &= \alpha + \beta_g \cdot \text{grade}_i + \beta_f \cdot \text{frequency}_i + \\
&\quad \beta_{graphe} \cdot \text{grapheme}_i + \beta_{grapho} \cdot \text{graphomotor}_i
\end{aligned}
$$

This kind of model is now able to predict valid number of stops (i.e., positive integers). Note that for simplicity, we omit the varying effects and the priors from the above model (for more details on Poisson regression, see Winter & Bürkner, 2021).

2.4 Fitting the final model

To set up the model, we need to invoke the `brms::brmsformula()` function and construct one formula for each of the four dependant variables. We fitted all models using the `brms` package (Bürkner, 2017) (for an introduction to Bayesian multilevel modelling in brms, see Nalborczyk, Batailler, Lœvenbruck, Vilain, & Bürkner, 2019). We used sum contrasts (i.e., recoding conditions as -0.5 vs. 0.5) for binary predictors (i.e., frequency, grapheme complexity, and graphomotor difficulty) and used the default factor coding scheme (i.e., dummy coding) for grade.

```
# defining the model formula for the generalised multilevel model
formula_generalised <-
```

```
    bf(
        duration ~ 1 + group * frequency * grapheme_complexity *
            graphomotor_difficulty + (1 | subject),
        family = shifted_lognormal()
        ) +
    bf(
        mean_velocity ~ 1 + group * frequency * grapheme_complexity *
            graphomotor_difficulty + (1 | subject),
        family = shifted_lognormal()
        ) +
    bf(
        number_of_stops ~ 1 + group * frequency * grapheme_complexity *
            graphomotor_difficulty + (1 | subject),
        family = poisson()
        ) +
    bf(
        letter_size ~ 1 + group * frequency * grapheme_complexity *
            graphomotor_difficulty + (1 | subject),
        family = shifted_lognormal()
        )

# defining the priors for the multilevel generalised model
priors_generalised <- c(
    prior(normal(1, 0.5), class = Intercept, resp = "duration"),
    prior(normal(0, 0.5), class = b, resp = "duration"),
    prior(exponential(0.1), class = sd, resp = "duration"),
    prior(exponential(0.1), class = sigma, resp = "duration"),
    prior(normal(2, 0.5), class = Intercept, resp = "meanvelocity"),
    prior(normal(0, 0.5), class = b, resp = "meanvelocity"),
    prior(exponential(0.1), class = sd, resp = "meanvelocity"),
    prior(exponential(0.1), class = sigma, resp = "meanvelocity"),
    prior(normal(1, 0.5), class = Intercept, resp = "numberofstops"),
    prior(normal(0, 0.5), class = b, resp = "numberofstops"),
    prior(exponential(0.1), class = sd, resp = "numberofstops"),
    prior(normal(2, 0.5), class = Intercept, resp = "lettersize"),
    prior(normal(0, 0.5), class = b, resp = "lettersize"),
```

```r
    prior(exponential(0.1), class = sd, resp = "lettersize"),
    prior(exponential(0.1), class = sigma, resp = "lettersize")
    )

# centering and reordering predictors
df2 <- df %>%
    mutate(
        group = factor(
            x = group,
            levels = c("CP", "CE", "CM"),
            labels = c("Grade1", "Grade3", "Grade5")
            ),
        frequency = factor(
            x = frequency,
            levels = c("LF", "HF"),
            labels = c("LF", "HF")
            ),
        grapheme_complexity = factor(
            x = grapheme_complexity,
            levels = c("Simple", "Complex"),
            labels = c("Simple", "Complex")
            ),
        graphomotor_difficulty = factor(
            x = graphomotor_difficulty,
            levels = c("EL", "HL"),
            labels = c("t", "f")
            )
        ) %>%
    # removes rows where duration is equal to 0
    filter(duration != 0)

# defining contrasts
contrasts(df2$frequency) <- c(-0.5, +0.5)
contrasts(df2$grapheme_complexity) <- c(-0.5, +0.5)
contrasts(df2$graphomotor_difficulty) <- c(-0.5, +0.5)

# fitting the model
```

```r
mod2 <- brm(
    formula = formula_generalised + set_rescor(rescor = FALSE),
    prior = priors_generalised,
    chains = 4, cores = 4,
    warmup = 2000, iter = 1e4,
    control = list(adapt_delta = 0.95),
    data = df2,
    sample_prior = TRUE,
    file = "models/multilevel_generalised_model"
    )
```

We then fit this model below using the `brms::brm()` function. We run four chains, each for 10000 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 $4 \times (10000 - 2000) = 32000$ samples from the (joint) posterior distribution that will be used for inference.

2.5 Evaluating the model

One way of evaluating the model is to evaluate its predictions. In Figure 4, we depict the distribution of the raw data along with the distribution of 100 simulated datasets (a posterior predictive check, as introduced previously).

As can be seen from Figure 4, 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.

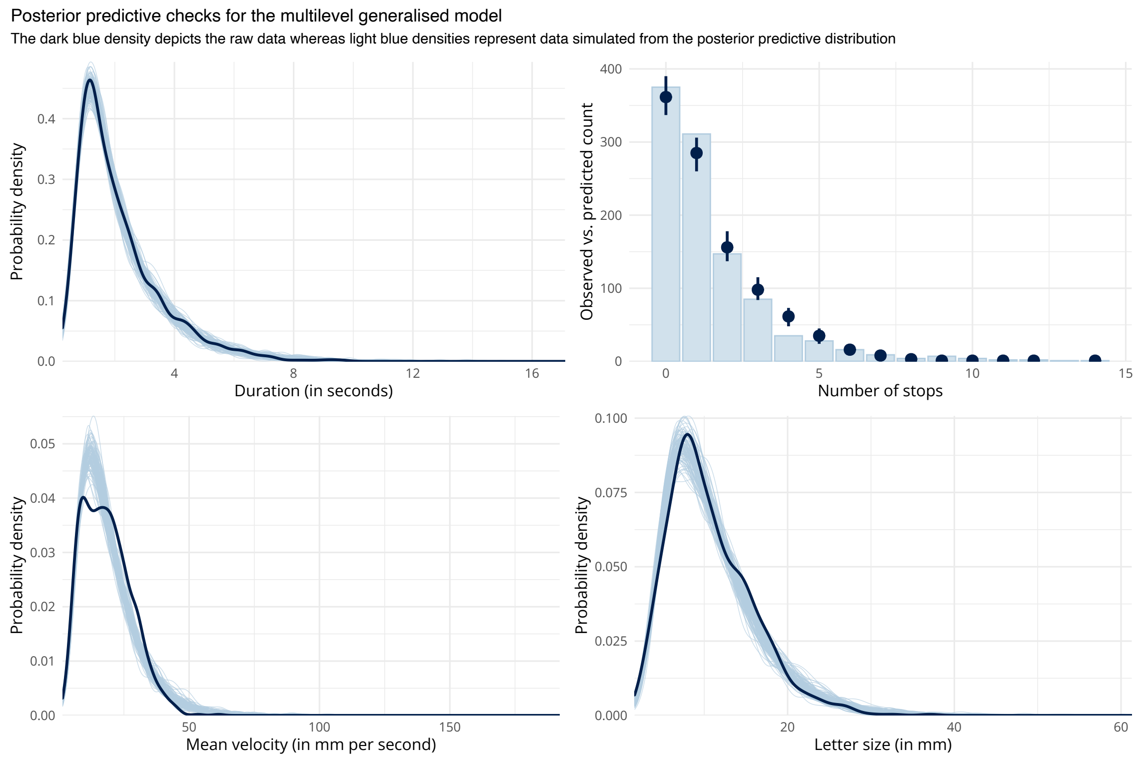

Figure 4. Posterior predictive checking. The dark blue line represents the distribution of raw data whereas light blue lines represent data simulated from the posterior distribution.

2.6 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). This method consists in comparing the posterior probability density to the prior probability density for some hypothesised value for the parameter of interest (e.g., $\theta = 0$). For instance, we test below the hypothesis according to which the effect of graphemic complexity in Grade 1 would be null.

```
# testing whether the effect of grapheme complexity on duration equal to 0
hyp <- hypothesis(x = mod2, hypothesis = "duration_grapheme_complexity1 = 0")

# prints the output
print(hyp)

## Hypothesis Tests for class b:
##                  Hypothesis Estimate Est.Error CI.Lower CI.Upper Evid.Ratio
## 1 (duration_graphem... = 0        0      0.05    -0.11      0.1       9.87
```

```
##   Post.Prob Star
## 1      0.91
## ---
## '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 = 2, alpha = 1) +
  geom_area(stat = "density", alpha = 0.8, position = "identity") +
  theme_bw(base_size = 12, base_family = "Open Sans") +
  labs(x = expression(beta[grapheme_complexity]), y = "Probability density") +
  scale_fill_brewer(palette = "Dark2") +
  theme(legend.title = element_blank() ) +
  coord_cartesian(xlim = c(-2, 2) )
```

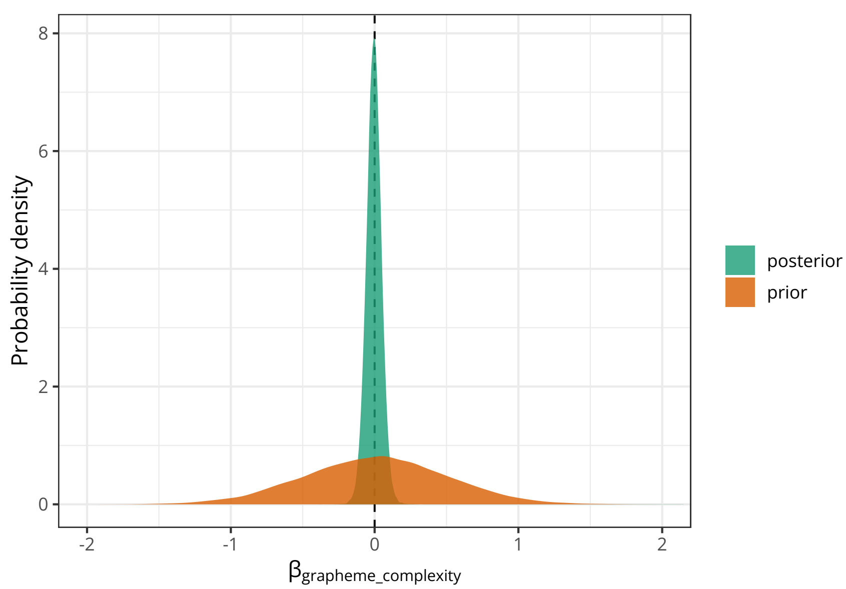

Figure 5. Hypothesis testing via the Savage-Dickey method. The resulting Bayes factor (BF) is the ratio of the height (i.e., the density probability) of the posterior versus prior distribution at some value of interest for the parameter (here it is 0).

The resulting Bayes factor (BF, called "Evid. Ratio" in the output) may be interpreted as follows: the observed data are 9.87 more likely under the hypothesis of null effect than under the hypothesis of a non-null effect. From the BF in favour of the null hypothesis (relative to the alternative hypothesis), we can compute the BF in favour of the alternative hypothesis (relative to the null hypothesis), using $\mathrm{BF}_{10} = 1/\mathrm{BF}_{01}$ (we report the BF_{10} in the following). 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 $\mathcal{H}_0$ versus $\mathcal{H}_1$) to convert them into *posterior odds* (the ratio of the a posteriori probability of $\mathcal{H}_0$ versus $\mathcal{H}_1$).

3 Interpretation of the results for each variable

Now that we have fitted the model, we are left with the task of interpreting the output from the model. 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. To summarise this distribution, we can retrieve samples from the joint posterior distribution.

```r
# retrieves posterior samples (for all parameters)
posterior_samples <- as_draws_df(mod2)

# displays a summary
posterior_summary <- summarise_draws(posterior_samples)

# displays the first six rows
head(posterior_summary)
```

```
## # A tibble: 6 x 10
##   variable        mean median     sd    mad     q5    q95  rhat ess_bulk ess_tail
##   <chr>          <dbl>  <dbl>  <dbl>  <dbl>  <dbl>  <dbl> <dbl>    <dbl>    <dbl>
## 1 b_duration_~   1.08   1.08  0.0310 0.0304  1.03   1.13   1.00   29023.   25012.
## 2 b_meanveloc~   2.34   2.34  0.0271 0.0263  2.29   2.38   1.00   30342.   20075.
## 3 b_numberofs~   1.04   1.04  0.0367 0.0354  0.977  1.10   1.00   41575.   26004.
## 4 b_lettersiz~   2.29   2.29  0.0234 0.0229  2.25   2.33   1.00   32779.   23505.
## 5 b_duration_~ -0.866 -0.865  0.0386 0.0384 -0.931 -0.804  1.00   23574.   24590.
## 6 b_duration_~ -0.967 -0.966  0.0414 0.0418 -1.04  -0.900  1.00   20887.   23486.
```

The above command 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. For instance, Figure 6 represents the posterior distribution of the average letter duration in Grade-1 children.

```r
# retrieves the posterior samples for the average letter duration in Grade 1
average_duration_grade1 <- posterior_samples$b_duration_Intercept +
    posterior_samples$ndt_duration

# plotting it
plotPost(
  paramSampleVec = exp(average_duration_grade1), showMode = TRUE,
```

```
xlab = expression(paste(alpha[duration][paste("[", Grade1, "]")] ) )
)
```

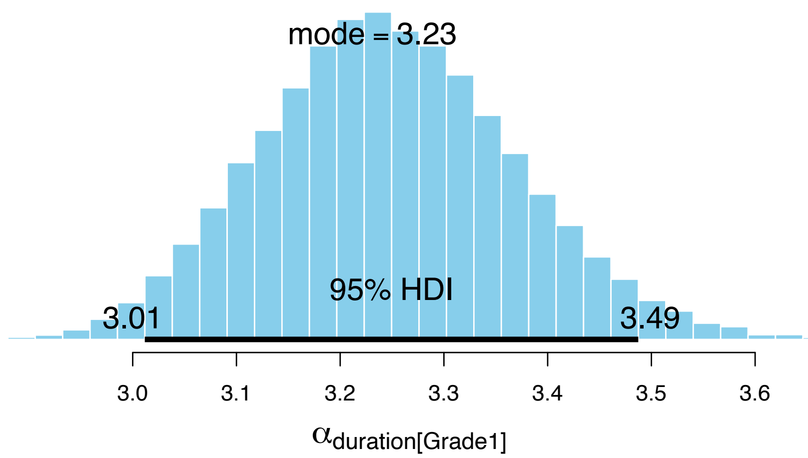

Figure 6. Posterior distribution of the intercept (i.e., the average letter duration in Grade 1). The mode (i.e., the most probable value) and the 95% credible (highest density) interval are also displayed.

Recall that we used a logarithmic link function, therefore the median letter duration is given by $\exp(\alpha + \text{shift})$.

3.1 Letter duration

Table 1 reports the estimates (median of the posterior distribution) and associated 95% credible intervals and BFs for all parameters regarding the letter duration variable.

Table 1
Estimates and BFs for the slopes for letter duration.

Term	Estimate	MAD	Lower	Upper	Rhat	BF10
Intercept	1.083	0.030	1.020	1.142	1.000	NA
groupGrade3	-0.865	0.038	-0.943	-0.793	1.000	7.281 x 10^17
groupGrade5	-0.966	0.042	-1.050	-0.889	1.000	3.586 x 10^15
frequency	-0.055	0.050	-0.157	0.049	1.000	0.192
grapheme_complexity	-0.005	0.050	-0.106	0.097	1.000	0.101
graphomotor_difficulty	0.423	0.051	0.320	0.526	1.000	9.762 x 10^15
groupGrade3:frequency	0.074	0.062	-0.049	0.196	1.000	0.258
groupGrade5:frequency	0.027	0.062	-0.092	0.148	1.000	0.136
groupGrade3:grapheme_complexity	0.011	0.062	-0.112	0.132	1.000	0.13
groupGrade5:grapheme_complexity	0.004	0.061	-0.115	0.124	1.000	0.124
frequency:grapheme_complexity	0.046	0.097	-0.151	0.244	1.000	0.225
groupGrade3:graphomotor_difficulty	0.001	0.063	-0.121	0.127	1.000	0.132
groupGrade5:graphomotor_difficulty	0.099	0.064	-0.024	0.225	1.000	0.436
frequency:graphomotor_difficulty	-0.071	0.097	-0.268	0.128	1.000	0.261
grapheme_complexity:graphomotor_difficulty	-0.008	0.098	-0.203	0.188	1.000	0.197
groupGrade3:frequency:grapheme_complexity	0.010	0.120	-0.229	0.248	1.000	0.249
groupGrade5:frequency:grapheme_complexity	-0.101	0.119	-0.337	0.131	1.000	0.352
groupGrade3:frequency:graphomotor_difficulty	0.007	0.122	-0.233	0.243	1.000	0.246
groupGrade5:frequency:graphomotor_difficulty	0.068	0.120	-0.167	0.308	1.000	0.285
groupGrade3:grapheme_complexity:graphomotor_difficulty	0.072	0.120	-0.162	0.309	1.000	0.291
groupGrade5:grapheme_complexity:graphomotor_difficulty	0.035	0.119	-0.201	0.270	1.000	0.252
frequency:grapheme_complexity:graphomotor_difficulty	-0.040	0.176	-0.401	0.321	1.000	0.368
groupGrade3:frequency:grapheme_complexity:graphomotor_difficulty	-0.006	0.217	-0.434	0.424	1.000	0.442
groupGrade5:frequency:grapheme_complexity:graphomotor_difficulty	0.134	0.213	-0.283	0.548	1.000	0.529

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 alternative hypothesis, relative to the null hypothesis (BF10).

These estimations are better understood visually. Thus, we plot the predictions of this model against raw data in Figure 7.

```
# retrieving the model's predictions
duration_predictions <- df2 %>%
    data_grid(graphomotor_difficulty, grapheme_complexity, frequency, group) %>%
    cbind(., fitted(
        object = mod2, newdata = ., resp = "duration",
        scale = "response", probs = c(0.025, 0.975),
        re_formula = NA, robust = TRUE
        ) ) %>%
    ungroup %>%
    dplyr::rename(estimate = Estimate, mad = Est.Error, lower = Q2.5, upper = Q97.5)
```

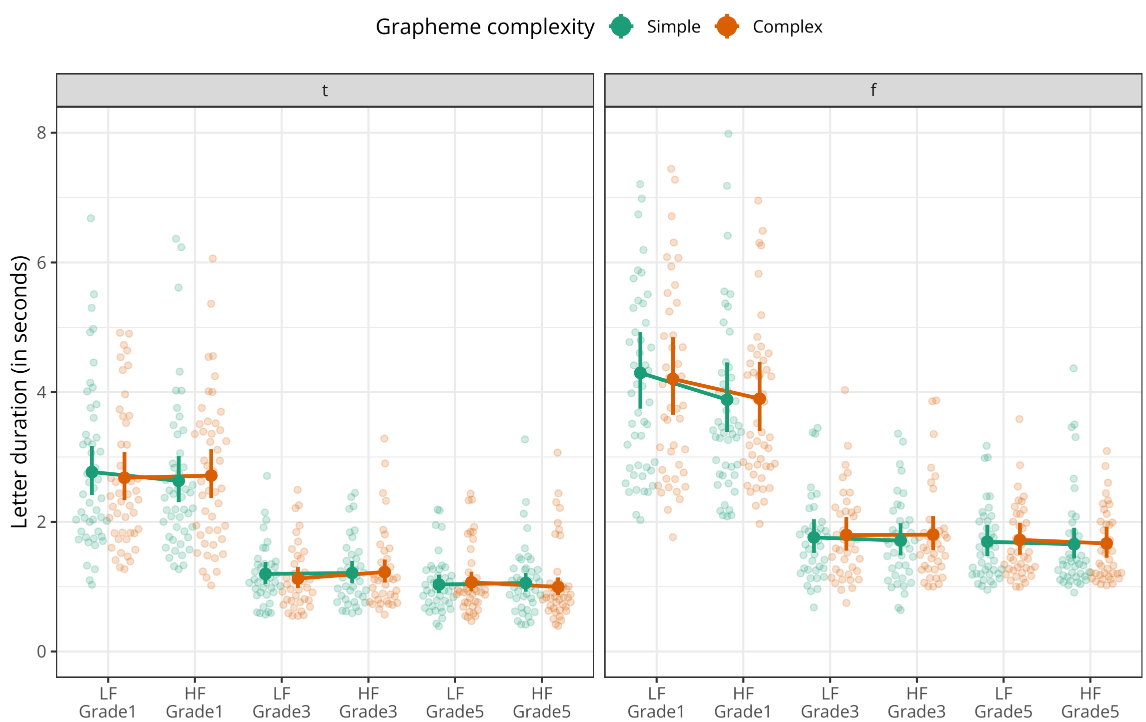

Figure 7. Letter duration by grade and word frequency (x-axis), grapheme complexity (in colour), and graphomotor difficulty (in panels). Transparent points represent individual data per participant. The surimposed dots and intervals represent the model's predictions (median and 95% credible interval of the posterior distribution).

As can be seen in Figure 7, the model predicts larger letter duration for the hard letter f as compared to the easy letter t for each grade. As can be seen from Table 1, the only BFs favouring the alternative hypothesis (relative to the null hypothesis) are the BFs for the difference between Grade 1 and Grade 3 in average letter duration (β = -0.865, 95% CrI [-0.943, -0.793], BF_{10} = 7.281 x 10^17), as well as the difference between Grade 1 and Grade 5 (β = -0.966, 95% CrI [-1.05, -0.889], BF_{10} = 3.586 x 10^15), and the effect of graphomotor difficulty in Grade 1 (β = 0.423, 95% CrI [0.32, 0.526], BF_{10} = 9.762 x 10^15). Predictions from this model for each condition are also summarised in Table 2.

Table 2
Estimated letter duration in each condition.

Group	Frequency	Grapheme complexity	Graphomotor difficulty	Estimate	MAD	Lower	Upper
Grade1	LF	Simple	t	2.768	0.186	2.416	3.172
Grade1	LF	Simple	f	4.296	0.287	3.745	4.923
Grade1	LF	Complex	t	2.680	0.179	2.334	3.077
Grade1	LF	Complex	f	4.203	0.291	3.651	4.846
Grade1	HF	Simple	t	2.632	0.176	2.303	3.011
Grade1	HF	Simple	f	3.883	0.263	3.387	4.458
Grade1	HF	Complex	t	2.713	0.182	2.367	3.122
Grade1	HF	Complex	f	3.899	0.261	3.401	4.469
Grade3	LF	Simple	t	1.196	0.084	1.040	1.381
Grade3	LF	Simple	f	1.760	0.127	1.522	2.040
Grade3	LF	Complex	t	1.127	0.080	0.980	1.304
Grade3	LF	Complex	f	1.797	0.127	1.556	2.074
Grade3	HF	Simple	t	1.211	0.085	1.053	1.399
Grade3	HF	Simple	f	1.710	0.121	1.482	1.980
Grade3	HF	Complex	t	1.226	0.087	1.063	1.418
Grade3	HF	Complex	f	1.802	0.129	1.561	2.090
Grade5	LF	Simple	t	1.033	0.070	0.902	1.185
Grade5	LF	Simple	f	1.693	0.120	1.470	1.953
Grade5	LF	Complex	t	1.068	0.072	0.931	1.230
Grade5	LF	Complex	f	1.722	0.121	1.493	1.985
Grade5	HF	Simple	t	1.057	0.072	0.923	1.213
Grade5	HF	Simple	f	1.654	0.115	1.435	1.909
Grade5	HF	Complex	t	0.996	0.066	0.869	1.142
Grade5	HF	Complex	f	1.667	0.117	1.447	1.924

Note. For each condition, the 'Estimate' and 'MAD' columns contain the median and the median absolute deviation (MAD) of the posterior distribution, respectively. The 'Lower' and 'Upper' columns contain the lower and upper bounds of the 95% credible interval.

The output of a Bayesian 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. To summarise this distribution, we can retrieve samples from the joint posterior distribution. Interestingly, this means we can look at the posterior distribution of any parameter of interest. For instance, and for exploratory purposes, we depict below the posterior distribution of the difference between high-frequency and low-frequency words (i.e., the effect of frequency) separately for each letter (graphomotor difficulty) and each grade. We averaged the predictions across both conditions of graphemic complexity, as this effect appeared to be null.

As can be seen in Figure 8, the posterior distribution for the effect of frequency is

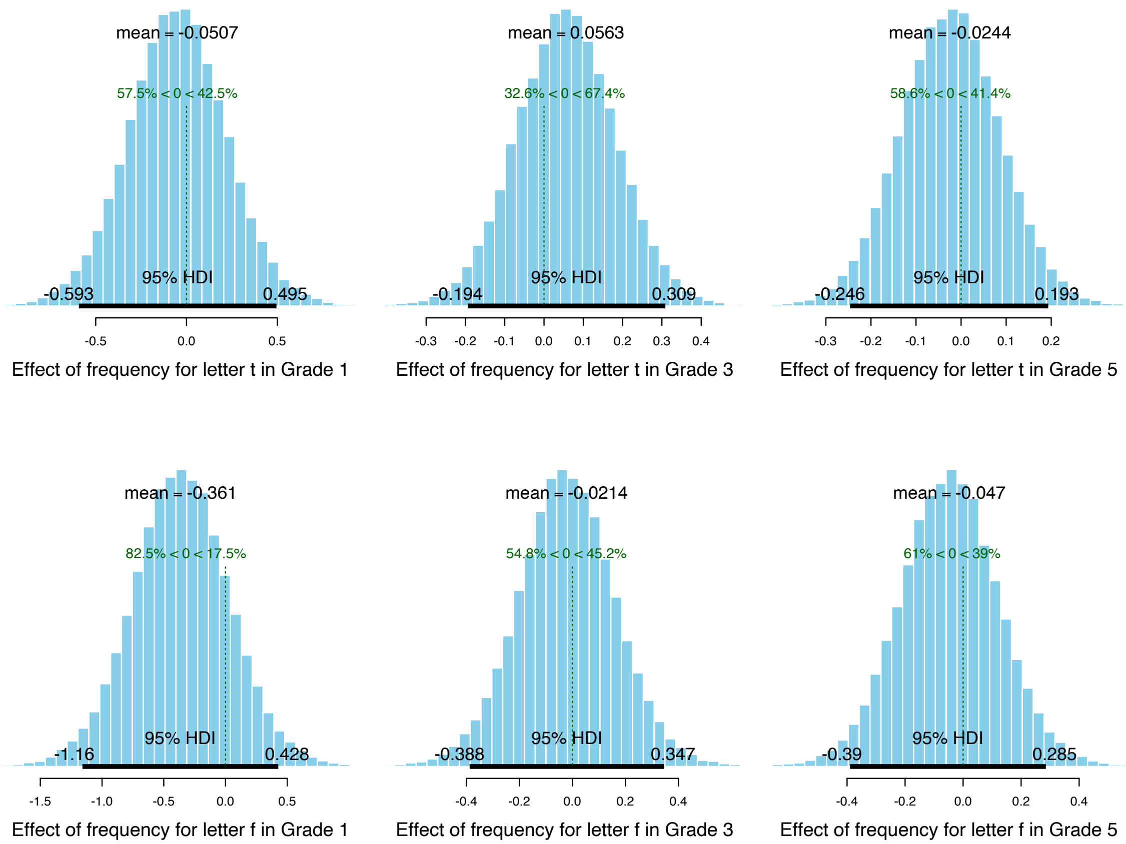

Figure 8. Effect of word frequency on letter duration (in seconds) for each grade (in column) and letter (in row). The histogram contains posterior samples for each effect, where the posterior distribution is summarised by its mean and 95% highest density interval (HDI). The green text indicates the probability that the parameter values is either inferior or superior to 0.

almost perfectly centred on zero in all conditions, except for letter f in Grade 1. Although the 95% credible interval largely encompasses 0 in this condition as well, there is still a 0.82 probability that the effect of frequency on letter duration is negative (given the data and the priors).

3.2 Number of stops

Table 3 reports the estimates (median of the posterior distribution) and associated 95% credible intervals and BFs for all parameters regarding the number of stops.

Table 3

Estimates and BFs for the slopes for the number of stops.

Term	Estimate	MAD	Lower	Upper	Rhat	BF10
Intercept	1.037	0.035	0.965	1.109	1.000	NA
groupGrade3	-1.384	0.073	-1.529	-1.243	1.000	4.244 x 10^16
groupGrade5	-1.556	0.077	-1.709	-1.409	1.000	6.510 x 10^15
frequency	-0.180	0.070	-0.324	-0.034	1.000	2.746
grapheme_complexity	0.031	0.070	-0.111	0.175	1.000	0.155
graphomotor_difficulty	-0.051	0.070	-0.192	0.094	1.000	0.184
groupGrade3:frequency	0.292	0.140	0.019	0.568	1.000	2.537
groupGrade5:frequency	0.128	0.147	-0.161	0.409	1.000	0.417
groupGrade3:grapheme_complexity	0.051	0.142	-0.227	0.329	1.000	0.298
groupGrade5:grapheme_complexity	-0.069	0.149	-0.356	0.220	1.000	0.334
frequency:grapheme_complexity	0.177	0.133	-0.096	0.446	1.000	0.653
groupGrade3:graphomotor_difficulty	0.112	0.140	-0.164	0.390	1.000	0.378
groupGrade5:graphomotor_difficulty	0.061	0.144	-0.226	0.346	1.000	0.313
frequency:graphomotor_difficulty	-0.312	0.134	-0.581	-0.032	1.000	3.307
grapheme_complexity:graphomotor_difficulty	-0.047	0.135	-0.315	0.224	1.000	0.285
groupGrade3:frequency:grapheme_complexity	-0.234	0.254	-0.731	0.258	1.000	0.791
groupGrade5:frequency:grapheme_complexity	-0.147	0.260	-0.653	0.358	1.000	0.602
groupGrade3:frequency:graphomotor_difficulty	0.102	0.257	-0.396	0.605	1.000	0.55
groupGrade5:frequency:graphomotor_difficulty	0.326	0.255	-0.176	0.834	1.000	1.153
groupGrade3:grapheme_complexity:graphomotor_difficulty	0.068	0.254	-0.427	0.568	1.000	0.529
groupGrade5:grapheme_complexity:graphomotor_difficulty	0.004	0.264	-0.511	0.509	1.000	0.52
frequency:grapheme_complexity:graphomotor_difficulty	0.042	0.231	-0.430	0.518	1.000	0.464
groupGrade3:frequency:grapheme_complexity:graphomotor_difficulty	-0.057	0.379	-0.801	0.678	1.000	0.766
groupGrade5:frequency:grapheme_complexity:graphomotor_difficulty	0.073	0.384	-0.679	0.836	1.000	0.782

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 alternative hypothesis, relative to the null hypothesis (BF10).

These estimations are better understood visually. Thus, we plot the predictions of this model against raw data in Figure 9.

```
# retrieving the model's predictions
stops_predictions <- df2 %>%
    data_grid(graphomotor_difficulty, grapheme_complexity, frequency, group) %>%
    cbind(., fitted(
        object = mod2, newdata = ., resp = "numberofstops",
        scale = "response", probs = c(0.025, 0.975),
        re_formula = NA, robust = TRUE
        ) ) %>%
    ungroup %>%
    dplyr::rename(estimate = Estimate, mad = Est.Error, lower = Q2.5, upper = Q97.5)
```

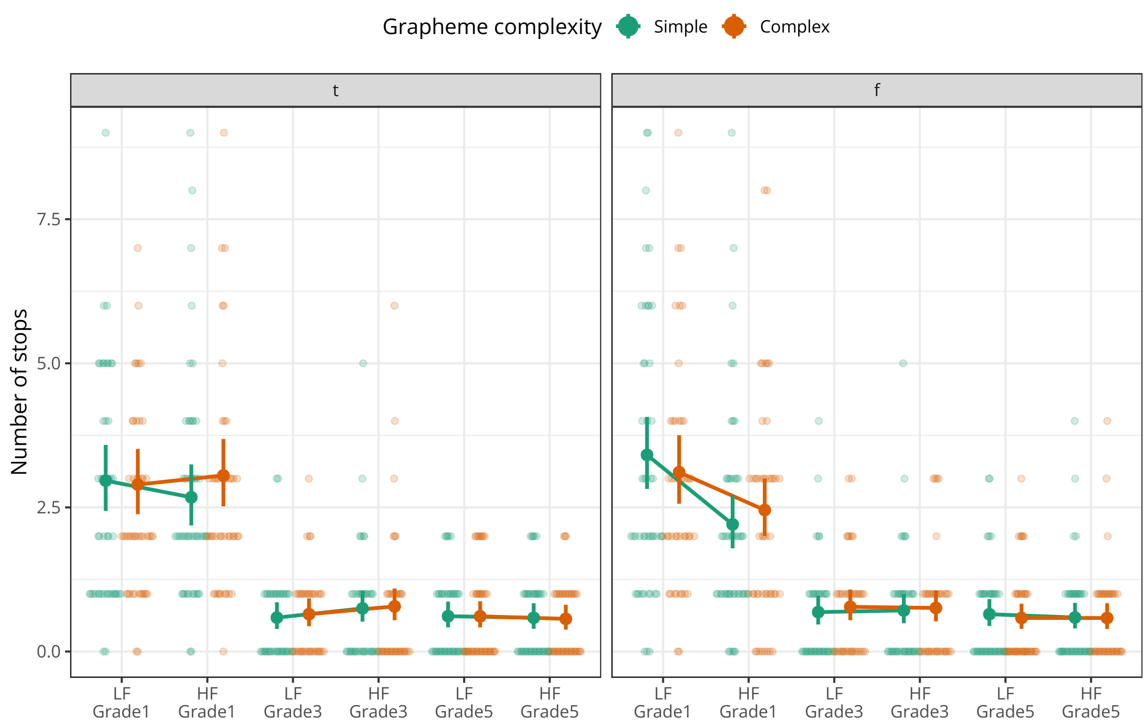

Figure 9. Number of stops by grade and word frequency (x-axis), grapheme complexity (in colour), and graphomotor difficulty (in panels). Transparent points represent individual data per participant. The surimposed dots and intervals represent the model's predictions (median and 95% credible interval of the posterior distribution).

As can be seen in Figure 9, the model most predicts an interaction between the effect of the word frequency and the effect of first-letter graphomotor difficulty in Grade 1, with infrequent words leading to a greater number of stops than frequent words for f (hard letter, HL) more than for t (easy letter, EL) (β = -0.312, 95% CrI [-0.581, -0.032], BF_{10} = 3.307). As can be seen from Table 3, others BFs favouring the alternative hypothesis (relative to the null hypothesis) are BFs for the difference between Grade 1 and Grade 3 (β = -1.384, 95% CrI [-1.529, -1.243], BF_{10} = 4.244 x 10^16), as well as between Grade 1 and Grade 5 (β = -1.556, 95% CrI [-1.709, -1.409], BF_{10} = 6.510 x 10^15), the effect of word frequency in Grade 1 (β = -0.18, 95% CrI [-0.324, -0.034], BF_{10} = 2.746), and Grade 3 (β = 0.292, 95% CrI [0.019, 0.568], BF_{10} = 2.537). Predictions from this model for each condition are also summarised in Table 4.

Table 4
Estimated number of stops in each condition.

Group	Frequency	Grapheme complexity	Graphomotor difficulty	Estimate	MAD	Lower	Upper
Grade1	LF	Simple	t	2.964	0.275	2.438	3.585
Grade1	LF	Simple	f	3.411	0.302	2.820	4.070
Grade1	LF	Complex	t	2.897	0.272	2.382	3.515
Grade1	LF	Complex	f	3.110	0.289	2.565	3.753
Grade1	HF	Simple	t	2.676	0.258	2.187	3.246
Grade1	HF	Simple	f	2.205	0.227	1.791	2.716
Grade1	HF	Complex	t	3.055	0.283	2.519	3.689
Grade1	HF	Complex	f	2.454	0.240	2.004	3.002
Grade3	LF	Simple	t	0.589	0.114	0.392	0.855
Grade3	LF	Simple	f	0.684	0.126	0.469	0.967
Grade3	LF	Complex	t	0.648	0.120	0.440	0.923
Grade3	LF	Complex	f	0.775	0.135	0.543	1.075
Grade3	HF	Simple	t	0.748	0.133	0.518	1.053
Grade3	HF	Simple	f	0.712	0.127	0.491	1.001
Grade3	HF	Complex	t	0.783	0.138	0.544	1.093
Grade3	HF	Complex	f	0.756	0.134	0.524	1.059
Grade5	LF	Simple	t	0.613	0.113	0.419	0.866
Grade5	LF	Simple	f	0.648	0.116	0.443	0.910
Grade5	LF	Complex	t	0.611	0.115	0.418	0.872
Grade5	LF	Complex	f	0.582	0.110	0.394	0.830
Grade5	HF	Simple	t	0.585	0.112	0.395	0.838
Grade5	HF	Simple	f	0.590	0.111	0.402	0.844
Grade5	HF	Complex	t	0.567	0.109	0.380	0.811
Grade5	HF	Complex	f	0.582	0.112	0.391	0.838

Note. For each condition, the 'Estimate' and 'MAD' columns contain the median and the median absolute deviation (MAD) of the posterior distribution, respectively. The 'Lower' and 'Upper' columns contain the lower and upper bounds of the 95% credible interval.

For exploratory purposes, we depict below the posterior distribution of the difference between high-frequency and low-frequency words (i.e., the effect of frequency) separately for each letter (graphomotor difficulty) and each grade. We averaged the predictions across both conditions of graphemic complexity, as this effect appeared to be null.

As can be seen in Figure 10, the posterior distribution for the effect of frequency is almost perfectly centred on zero in all conditions, except for letter f in Grade 1. In this condition, the 95% credible interval excludes 0 and there is a 0.98 probability that the effect

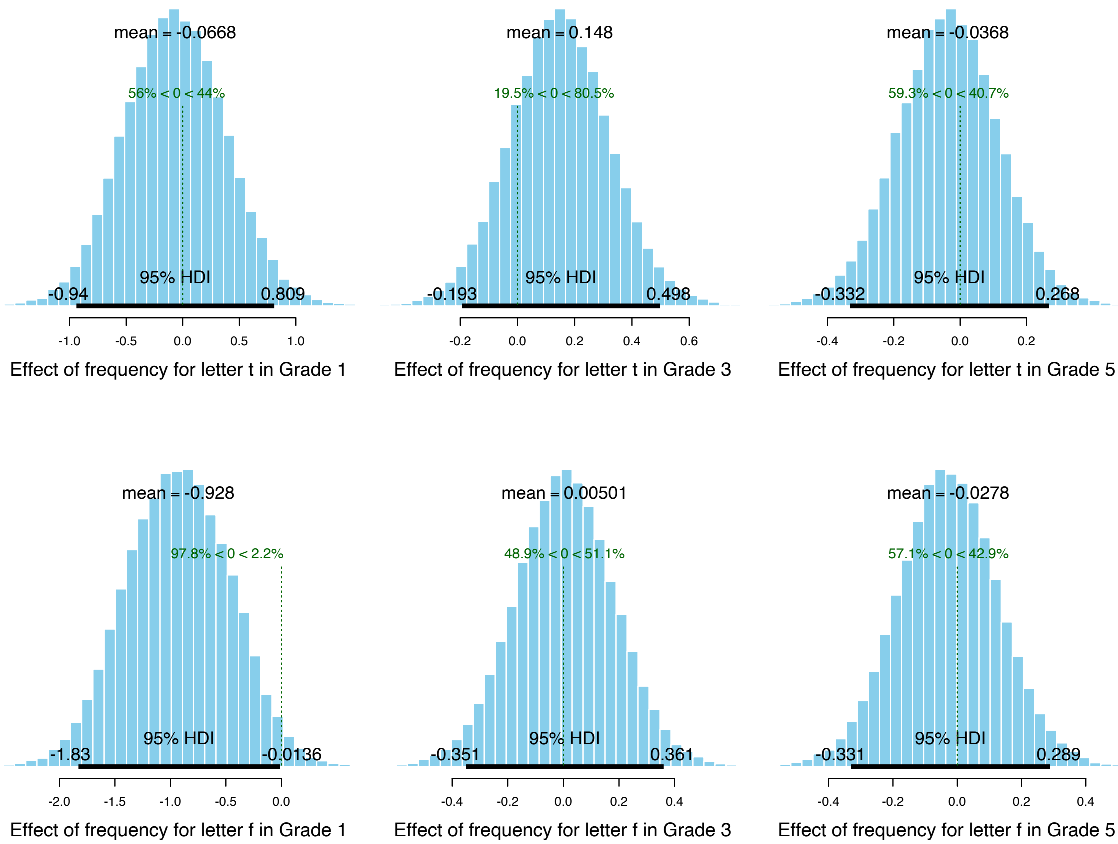

Figure 10. Effect of word frequency on the number of stops for each grade (in column) and letter (in row). The histogram contains posterior samples for each effect, where the posterior distribution is summarised by its mean and 95% highest density interval (HDI). The green text indicates the probability that the parameter values is either inferior or superior to 0.

of frequency on the number of stops is negative (given the data and the priors).

3.3 Mean velocity

Table 5 reports the estimates (median of the posterior distribution) and associated 95% credible intervals and BFs for all parameters regarding the mean velocity.

Table 5

Estimates and BFs for the slopes for the mean velocity.

Term	Estimate	MAD	Lower	Upper	Rhat	BF10
Intercept	2.338	0.026	2.281	2.387	1.000	NA
groupGrade3	0.562	0.033	0.496	0.628	1.000	1.213 x 10^15
groupGrade5	0.793	0.033	0.729	0.859	1.000	1.859 x 10^16
frequency	0.049	0.046	-0.043	0.141	1.000	0.162
grapheme_complexity	0.036	0.047	-0.057	0.130	1.000	0.123
graphomotor_difficulty	0.159	0.046	0.066	0.252	1.000	16.037
groupGrade3:frequency	-0.063	0.065	-0.188	0.064	1.000	0.199
groupGrade5:frequency	-0.031	0.064	-0.156	0.095	1.000	0.142
groupGrade3:grapheme_complexity	-0.049	0.066	-0.177	0.080	1.000	0.169
groupGrade5:grapheme_complexity	-0.025	0.065	-0.151	0.100	1.000	0.137
frequency:grapheme_complexity	-0.051	0.089	-0.232	0.129	1.000	0.208
groupGrade3:graphomotor_difficulty	-0.025	0.066	-0.154	0.103	1.000	0.137
groupGrade5:graphomotor_difficulty	-0.031	0.064	-0.156	0.095	1.000	0.14
frequency:graphomotor_difficulty	0.158	0.091	-0.024	0.337	1.000	0.78
grapheme_complexity:graphomotor_difficulty	0.027	0.091	-0.154	0.206	1.000	0.189
groupGrade3:frequency:grapheme_complexity	0.015	0.125	-0.229	0.261	1.000	0.248
groupGrade5:frequency:grapheme_complexity	0.097	0.123	-0.144	0.341	1.000	0.342
groupGrade3:frequency:graphomotor_difficulty	-0.049	0.127	-0.295	0.198	1.000	0.268
groupGrade5:frequency:graphomotor_difficulty	-0.174	0.124	-0.412	0.069	1.000	0.635
groupGrade3:grapheme_complexity:graphomotor_difficulty	-0.036	0.125	-0.284	0.213	1.000	0.255
groupGrade5:grapheme_complexity:graphomotor_difficulty	-0.020	0.124	-0.261	0.224	1.000	0.25
frequency:grapheme_complexity:graphomotor_difficulty	0.070	0.166	-0.249	0.397	1.000	0.357
groupGrade3:frequency:grapheme_complexity:graphomotor_difficulty	0.044	0.227	-0.395	0.487	1.000	0.457
groupGrade5:frequency:grapheme_complexity:graphomotor_difficulty	-0.086	0.222	-0.523	0.351	1.000	0.465

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 alternative hypothesis, relative to the null hypothesis (BF10).

These estimations are better understood visually. Thus, we plot the predictions of this model against raw data in Figure 11.

```
# retrieving the model's predictions
velocity_predictions <- df2 %>%
    data_grid(graphomotor_difficulty, grapheme_complexity, frequency, group) %>%
    cbind(., fitted(
        object = mod2, newdata = ., resp = "meanvelocity",
        scale = "response", probs = c(0.025, 0.975),
        re_formula = NA, robust = TRUE
        ) ) %>%
    ungroup %>%
    dplyr::rename(estimate = Estimate, mad = Est.Error, lower = Q2.5, upper = Q97.5)
```

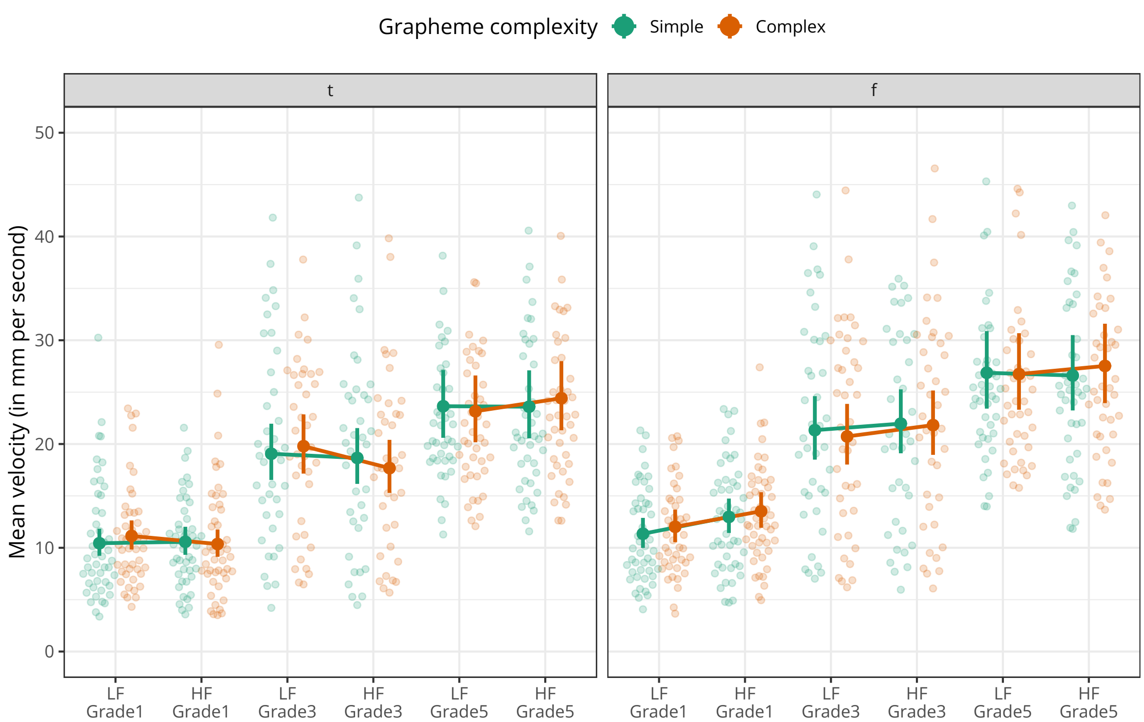

Figure 11. Mean velocity by grade and word frequency (x-axis), grapheme complexity (in colour), and graphomotor difficulty (in panels). Transparent points represent individual data per participant. The surimposed dots and intervals represent the model's predictions (median and 95% credible interval of the posterior distribution).

As can be seen in Figure 11, the model most notably predicts higher velocity for the hard letter f as compared to the easy letter t, excepted for low frequency words in Grade 1.

First graders seem to have lower velocity than third graders, who themselves seem to have lower velocity than fifth graders on average. As can be seen from Table 5, BFs favouring the alternative hypothesis (relative to the null hypothesis) are BFs for the difference between Grade 1 and Grade 3 ($\beta = 0.562$, 95% CrI [0.496, 0.628], $BF_{10} = 1.213$ x 10^15), as well as between Grade 1 and Grade 5 ($\beta = 0.793$, 95% CrI [0.729, 0.859], $BF_{10} = 1.859$ x 10^16), and the effect of graphomotor difficulty in Grade 1 ($\beta = 0.159$, 95% CrI [0.066, 0.252], $BF_{10} = 16.037$). Predictions from this model for each condition are also summarised in Table 6.

Table 6
Estimated mean velocity in each condition.

Group	Frequency	Grapheme complexity	Graphomotor difficulty	Estimate	MAD	Lower	Upper
Grade1	LF	Simple	t	10.439	0.649	9.228	11.835
Grade1	LF	Simple	f	11.350	0.730	9.992	12.881
Grade1	LF	Complex	t	11.140	0.708	9.836	12.636
Grade1	LF	Complex	f	12.020	0.793	10.538	13.677
Grade1	HF	Simple	t	10.576	0.676	9.321	12.025
Grade1	HF	Simple	f	12.975	0.831	11.427	14.742
Grade1	HF	Complex	t	10.362	0.661	9.131	11.763
Grade1	HF	Complex	f	13.521	0.855	11.928	15.358
Grade3	LF	Simple	t	19.061	1.367	16.534	21.957
Grade3	LF	Simple	f	21.336	1.525	18.496	24.615
Grade3	LF	Complex	t	19.793	1.427	17.149	22.855
Grade3	LF	Complex	f	20.730	1.487	18.022	23.867
Grade3	HF	Simple	t	18.659	1.332	16.159	21.537
Grade3	HF	Simple	f	21.965	1.535	19.107	25.269
Grade3	HF	Complex	t	17.680	1.288	15.295	20.402
Grade3	HF	Complex	f	21.822	1.544	18.957	25.160
Grade5	LF	Simple	t	23.643	1.627	20.601	27.133
Grade5	LF	Simple	f	26.864	1.876	23.411	30.888
Grade5	LF	Complex	t	23.158	1.610	20.167	26.606
Grade5	LF	Complex	f	26.743	1.868	23.308	30.681
Grade5	HF	Simple	t	23.599	1.649	20.547	27.100
Grade5	HF	Simple	f	26.604	1.844	23.236	30.499
Grade5	HF	Complex	t	24.416	1.695	21.327	27.982
Grade5	HF	Complex	f	27.526	1.937	23.944	31.594

Note. For each condition, the 'Estimate' and 'MAD' columns contain the median and the median absolute deviation (MAD) of the posterior distribution, respectively. The 'Lower' and 'Upper' columns contain the lower and upper bounds of the 95% credible interval.

For exploratory purposes, we depict below the posterior distribution of the difference between high-frequency and low-frequency words (i.e., the effect of frequency) separately for each letter (graphomotor difficulty) and each grade. We averaged the predictions across both conditions of graphemic complexity, as this effect appeared to be null.

As can be seen in Figure 12, the posterior distribution for the effect of frequency is almost perfectly centred on zero in all conditions, except for letter f in Grade 1. In this condition, there is a 0.92 probability that the effect of frequency on the mean velocity (in mm per second) is positive (given the data and the priors).

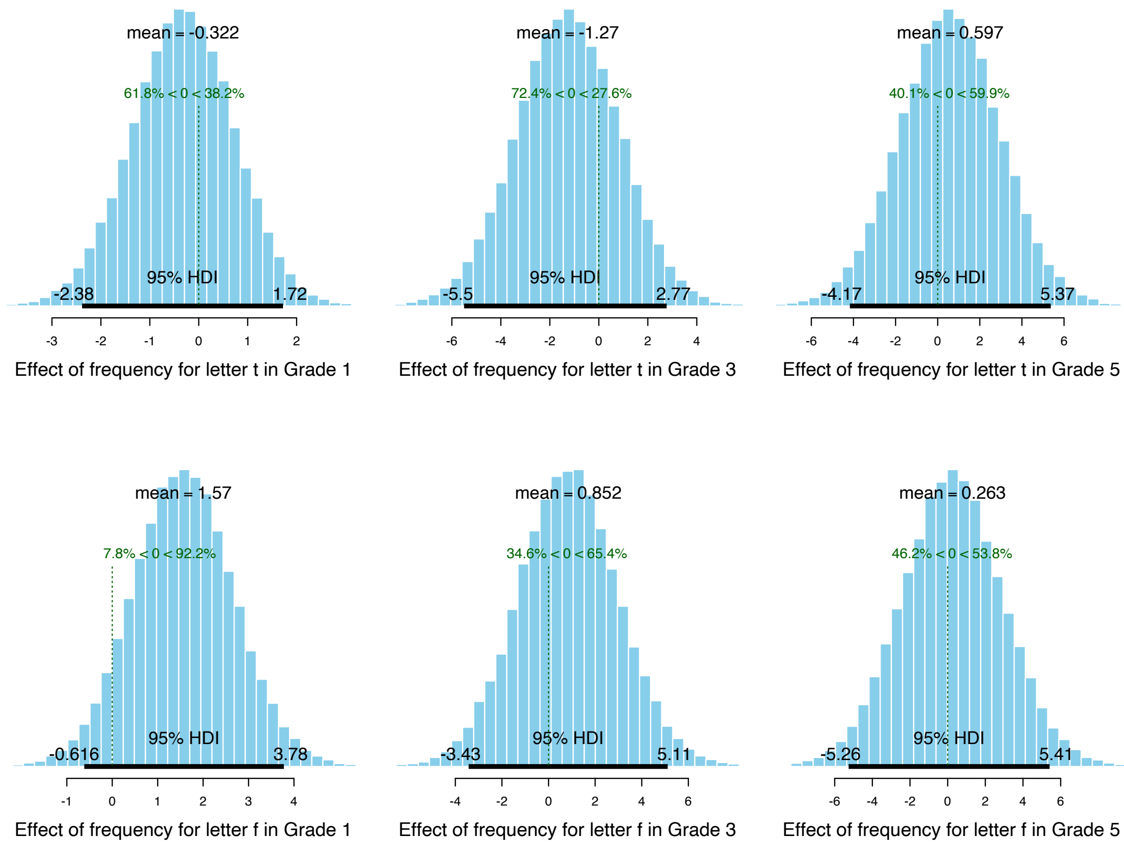

Figure 12. Effect of word frequency on the mean velocity (in mm per second) for each grade (in column) and letter (in row). The histogram contains posterior samples for each effect, where the posterior distribution is summarised by its mean and 95% highest density interval (HDI). The green text indicates the probability that the parameter values is either inferior or superior to 0.

3.4 Letter size

Table 7 reports the estimates (median of the posterior distribution) and associated 95% credible intervals and BFs for all parameters regarding the letter size.

Table 7

Estimates and BFs for the slopes for the letter size.

Term	Estimate	MAD	Lower	Upper	Rhat	BF10
Intercept	2.289	0.023	2.241	2.333	1.000	NA
groupGrade3	-0.125	0.029	-0.181	-0.070	1.000	7.090 x 10^3
groupGrade5	0.006	0.028	-0.049	0.061	1.000	0.059
frequency	0.023	0.042	-0.059	0.107	1.000	0.097
grapheme_complexity	0.032	0.042	-0.051	0.116	1.000	0.115
graphomotor_difficulty	0.703	0.042	0.618	0.788	1.000	-9.629 x 10^16
groupGrade3:frequency	-0.021	0.057	-0.133	0.090	1.000	0.124
groupGrade5:frequency	-0.004	0.055	-0.112	0.105	1.000	0.112
groupGrade3:grapheme_complexity	-0.027	0.056	-0.138	0.085	1.000	0.126
groupGrade5:grapheme_complexity	-0.012	0.055	-0.121	0.097	1.000	0.111
frequency:grapheme_complexity	0.023	0.081	-0.139	0.186	1.000	0.17
groupGrade3:graphomotor_difficulty	-0.115	0.057	-0.225	-0.003	1.000	0.843
groupGrade5:graphomotor_difficulty	-0.193	0.055	-0.303	-0.084	1.000	42.547
frequency:graphomotor_difficulty	0.074	0.081	-0.090	0.235	1.000	0.251
grapheme_complexity:graphomotor_difficulty	0.022	0.081	-0.142	0.186	1.000	0.169
groupGrade3:frequency:grapheme_complexity	0.004	0.109	-0.208	0.220	1.000	0.222
groupGrade5:frequency:grapheme_complexity	-0.048	0.108	-0.260	0.164	1.000	0.237
groupGrade3:frequency:graphomotor_difficulty	-0.041	0.110	-0.257	0.177	1.000	0.237
groupGrade5:frequency:graphomotor_difficulty	-0.102	0.108	-0.316	0.108	1.000	0.334
groupGrade3:grapheme_complexity:graphomotor_difficulty	0.012	0.111	-0.207	0.228	1.000	0.228
groupGrade5:grapheme_complexity:graphomotor_difficulty	0.030	0.109	-0.185	0.244	1.000	0.229
frequency:grapheme_complexity:graphomotor_difficulty	0.053	0.150	-0.248	0.352	1.000	0.324
groupGrade3:frequency:grapheme_complexity:graphomotor_difficulty	-0.025	0.204	-0.423	0.370	1.000	0.408
groupGrade5:frequency:grapheme_complexity:graphomotor_difficulty	0.023	0.201	-0.369	0.414	1.000	0.409

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 alternative hypothesis, relative to the null hypothesis (BF10).

These estimations are better understood visually. Thus, we plot the predictions of this model against raw data in Figure 13.

```
# retrieving the model's predictions
size_predictions <- df2 %>%
    data_grid(graphomotor_difficulty, grapheme_complexity, frequency, group) %>%
    cbind(., fitted(
        object = mod2, newdata = ., resp = "lettersize",
        scale = "response", probs = c(0.025, 0.975),
        re_formula = NA, robust = TRUE
        ) ) %>%
    ungroup %>%
    dplyr::rename(estimate = Estimate, mad = Est.Error, lower = Q2.5, upper = Q97.5)
```

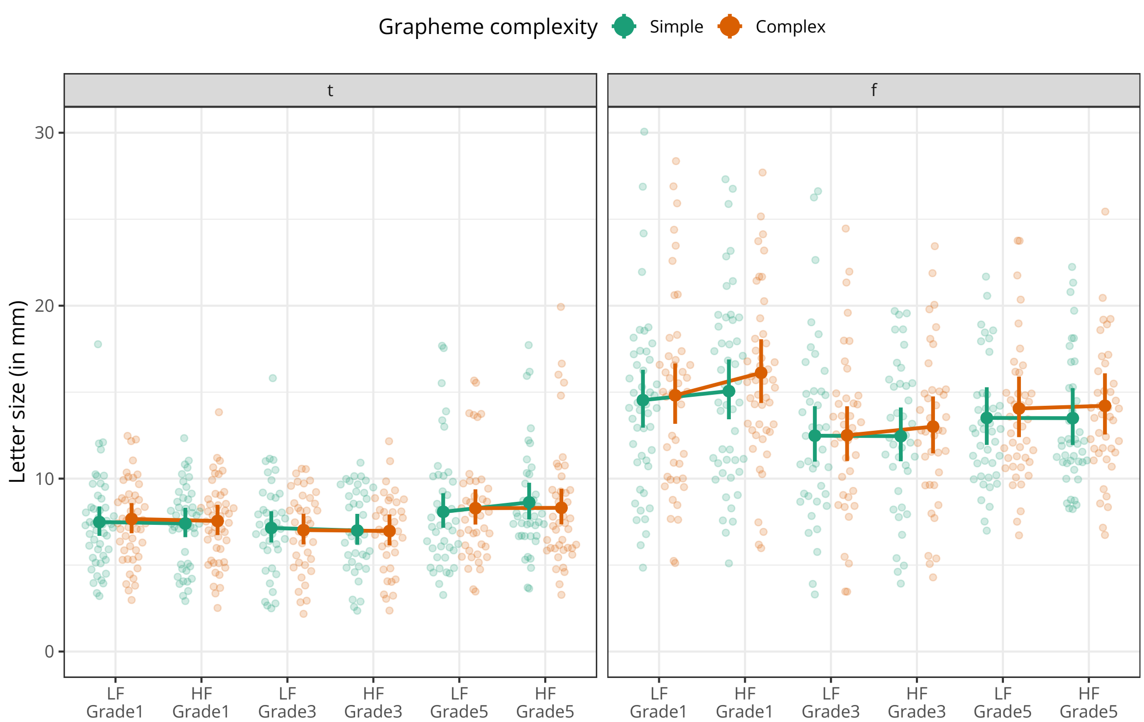

Figure 13. Letter size by grade and word frequency (x-axis), grapheme complexity (in colour), and graphomotor difficulty (in panels). Transparent points represent individual data per participant. The surimposed dots and intervals represent the model's predictions (median and 95% credible interval of the posterior distribution).

As can be seen in Figure 13, the production of difficult letters was associated with greater letter size than the production of easy letters for all grades. As can be seen from

Table 7, BFs favouring the alternative hypothesis (relative to the null hypothesis) are BFs for the difference between Grade 1 and Grade 3 (β = -0.125, 95% CrI [-0.181, -0.07], BF_{10} = 7.090 x 10^3), the effect of graphomotor difficulty in Grade 1 (β = 0.703, 95% CrI [0.618, 0.788], BF_{10} = -9.629 x 10^16), and the effect of graphomotor difficulty in Grade 5 (β = -0.193, 95% CrI [-0.303, -0.084], BF_{10} = 42.547). Predictions from this model for each condition are also summarised in Table 8.

Table 8
Estimated letter size in each condition.

Group	Frequency	Grapheme complexity	Graphomotor difficulty	Estimate	MAD	Lower	Upper
Grade1	LF	Simple	t	7.489	0.427	6.691	8.391
Grade1	LF	Simple	f	14.538	0.837	12.946	16.292
Grade1	LF	Complex	t	7.665	0.436	6.838	8.586
Grade1	LF	Complex	f	14.815	0.872	13.169	16.659
Grade1	HF	Simple	t	7.405	0.429	6.614	8.310
Grade1	HF	Simple	f	15.058	0.852	13.431	16.889
Grade1	HF	Complex	t	7.550	0.430	6.735	8.483
Grade1	HF	Complex	f	16.120	0.922	14.372	18.056
Grade3	LF	Simple	t	7.147	0.456	6.305	8.115
Grade3	LF	Simple	f	12.482	0.793	10.983	14.186
Grade3	LF	Complex	t	7.022	0.449	6.197	7.969
Grade3	LF	Complex	f	12.503	0.787	11.021	14.182
Grade3	HF	Simple	t	6.998	0.434	6.180	7.962
Grade3	HF	Simple	f	12.454	0.786	11.002	14.111
Grade3	HF	Complex	t	6.972	0.446	6.139	7.925
Grade3	HF	Complex	f	13.006	0.832	11.461	14.752
Grade5	LF	Simple	t	8.081	0.498	7.157	9.157
Grade5	LF	Simple	f	13.509	0.824	11.949	15.281
Grade5	LF	Complex	t	8.288	0.506	7.343	9.378
Grade5	LF	Complex	f	14.054	0.873	12.399	15.896
Grade5	HF	Simple	t	8.625	0.525	7.638	9.761
Grade5	HF	Simple	f	13.495	0.827	11.942	15.235
Grade5	HF	Complex	t	8.310	0.515	7.359	9.412
Grade5	HF	Complex	f	14.209	0.884	12.551	16.088

Note. For each condition, the 'Estimate' and 'MAD' columns contain the median and the median absolute deviation (MAD) of the posterior distribution, respectively. The 'Lower' and 'Upper' columns contain the lower and upper bounds of the 95% credible interval.

For exploratory purposes, we depict below the posterior distribution of the difference between high-frequency and low-frequency words (i.e., the effect of frequency) separately for each letter (graphomotor difficulty) and each grade. We averaged the predictions across both conditions of graphemic complexity, as this effect appeared to be null.

As can be seen in Figure 14, the posterior distribution for the effect of frequency is almost perfectly centred on zero in all conditions, except for letter f in Grade 1. In this condition, there is a 0.76 probability that the effect of frequency on the letter size (in mm) is positive (given the data and the priors).

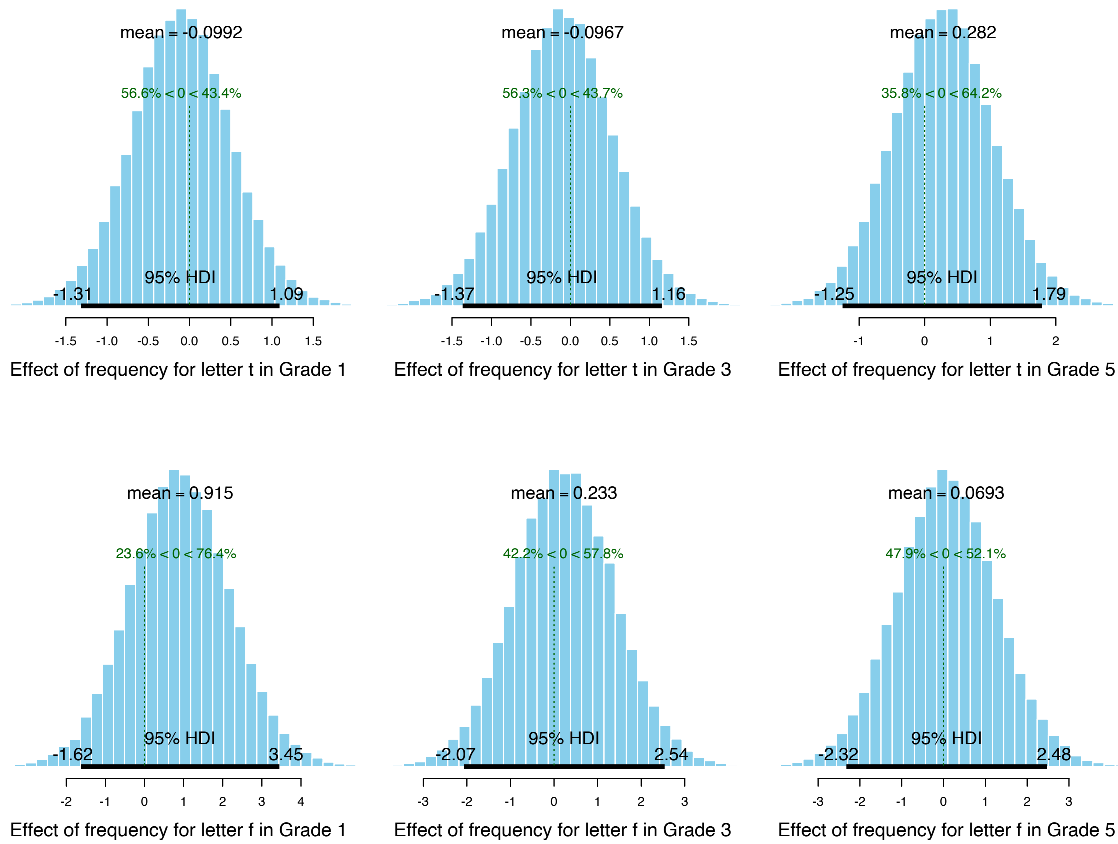

Figure 14. Effect of word frequency on the letter size (in mm) for each grade (in column) and letter (in row). The histogram contains posterior samples for each effect, where the posterior distribution is summarised by its mean and 95% highest density interval (HDI). The green text indicates the probability that the parameter values is either inferior or superior to 0.

4 Acknowledgments

This work, carried out within the Labex BLRI (ANR-11-LABX-0036) and the Institut Convergence ILCB (ANR-16-CONV-0002), has benefited from support from the French Government, managed by the French National Agency for Research (ANR), under the project title DYSTACMAP (ANR-13-APPR-0010).

5 Session information

```r
sessionInfo()
```

```
## R version 4.1.1 (2021-08-10)
## Platform: x86_64-apple-darwin17.0 (64-bit)
## Running under: macOS Big Sur 10.16
##
## Matrix products: default
## BLAS:   /Library/Frameworks/R.framework/Versions/4.1/Resources/lib/libRblas.0.dylib
## LAPACK: /Library/Frameworks/R.framework/Versions/4.1/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] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
##  [1] brms_2.16.2       Rcpp_1.0.7        BEST_0.5.3        HDInterval_0.2.2
##  [5] glue_1.4.2        knitr_1.36        papaja_0.1.0.9997 readxl_1.3.1
##  [9] GGally_2.1.2      modelr_0.1.8      tidybayes_3.0.1   posterior_1.1.0
## [13] patchwork_1.1.1   forcats_0.5.1     stringr_1.4.0     dplyr_1.0.7
## [17] purrr_0.3.4       readr_2.0.2       tidyr_1.1.4       tibble_3.1.5
## [21] tidyverse_1.3.1   ggbeeswarm_0.6.0  ggplot2_3.3.5     extraDistr_1.9.1
##
## loaded via a namespace (and not attached):
##   [1] backports_1.2.1      plyr_1.8.6           igraph_1.2.6
##   [4] splines_4.1.1        svUnit_1.0.6         crosstalk_1.1.1
##   [7] rstantools_2.1.1     inline_0.3.19        digest_0.6.28
##  [10] htmltools_0.5.2      rsconnect_0.8.24     fansi_0.5.0
##  [13] magrittr_2.0.1       checkmate_2.0.0      tzdb_0.1.2
##  [16] RcppParallel_5.1.4   matrixStats_0.61.0   xts_0.12.1
##  [19] prettyunits_1.1.1    colorspace_2.0-2     rvest_1.0.1
##  [22] ggdist_3.0.0         haven_2.4.3          xfun_0.26
##  [25] callr_3.7.0          crayon_1.4.1         jsonlite_1.7.2
##  [28] lme4_1.1-27.1        zoo_1.8-9            gtable_0.3.0
```

```
##  [31] emmeans_1.7.0        V8_3.4.2             distributional_0.2.2
##  [34] pkgbuild_1.2.0       rstan_2.26.3         abind_1.4-5
##  [37] scales_1.1.1         mvtnorm_1.1-3        DBI_1.1.1
##  [40] miniUI_0.1.1.1       xtable_1.8-4         stats4_4.1.1
##  [43] StanHeaders_2.26.3   DT_0.19              htmlwidgets_1.5.4
##  [46] httr_1.4.2           threejs_0.3.3        arrayhelpers_1.1-0
##  [49] RColorBrewer_1.1-2   ellipsis_0.3.2       pkgconfig_2.0.3
##  [52] reshape_0.8.8        loo_2.4.1            farver_2.1.0
##  [55] dbplyr_2.1.1         utf8_1.2.2           labeling_0.4.2
##  [58] tidyselect_1.1.1     rlang_0.4.11         reshape2_1.4.4
##  [61] later_1.3.0          munsell_0.5.0        cellranger_1.1.0
##  [64] tools_4.1.1          cli_3.0.1            generics_0.1.0
##  [67] broom_0.7.9          ggridges_0.5.3       evaluate_0.14
##  [70] fastmap_1.1.0        yaml_2.2.1           processx_3.5.2
##  [73] fs_1.5.0             nlme_3.1-152         projpred_2.0.2
##  [76] mime_0.12            xml2_1.3.2           compiler_4.1.1
##  [79] bayesplot_1.8.1      shinythemes_1.2.0    rstudioapi_0.13
##  [82] gamm4_0.2-6          beeswarm_0.4.0       curl_4.3.2
##  [85] reprex_2.0.1         stringi_1.7.5        ps_1.6.0
##  [88] Brobdingnag_1.2-6    lattice_0.20-44      Matrix_1.3-4
##  [91] nloptr_1.2.2.2       markdown_1.1         shinyjs_2.0.0
##  [94] tensorA_0.36.2       vctrs_0.3.8          pillar_1.6.3
##  [97] lifecycle_1.0.1      bridgesampling_1.1-2 estimability_1.3
## [100] httpuv_1.6.3         R6_2.5.1             bookdown_0.24
## [103] promises_1.2.0.1     gridExtra_2.3        vipor_0.4.5
## [106] rjags_4-11           codetools_0.2-18     boot_1.3-28
## [109] MASS_7.3-54          colourpicker_1.1.1   gtools_3.9.2
## [112] assertthat_0.2.1     withr_2.4.2          shinystan_2.5.0
## [115] mgcv_1.8-36          parallel_4.1.1       hms_1.1.1
## [118] grid_4.1.1           minqa_1.2.4          coda_0.19-4
## [121] rmarkdown_2.11       shiny_1.7.1          lubridate_1.8.0
## [124] base64enc_0.1-3      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

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

Gelman, A., Vehtari, A., Simpson, D., Margossian, C. C., Carpenter, B., Yao, Y., … Modrák, M. (2020). Bayesian workflow. *arXiv:2011.01808 [Stat].* Retrieved from http://arxiv.org/abs/2011.01808

Nalborczyk, L., Batailler, C., Lœvenbruck, H., Vilain, A., & Bürkner, P.-C. (2019). An introduction to bayesian multilevel models using brms: A case study of gender effects on vowel variability in standard indonesian. *Journal of Speech Language and Hearing Research*, *62*(5), 1225–1242. https://doi.org/10.1044/2018_JSLHR-S-18-0006

R Core Team. (2021). *R: A language and environment for statistical computing.* Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

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

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

Winter, B., & Bürkner, P.-. C. (2021). *Poisson regression for linguists: A tutorial introduction to modeling count data with brms.* preprint, preprint. https://doi.org/10.31219/osf.io/93kaf

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